2019

Interactive Perception in Robotics

Masoud Baghbahari Baghdadabad
University of Central Florida

INTERACTIVE PERCEPTION IN ROBOTICS

by

MASOUD BAGHBAHARI BAGHDADABAD
M.S. Sharif University of Technology, 2013

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical and Computer Engineering
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2019

Major Professor: Aman Behal

ABSTRACT

Interactive perception is a significant and unique characteristic of embodied agents. An agent can discover plenty of knowledge through active interaction with its surrounding environment. Recently, deep learning structures introduced new possibilities to interactive perception in robotics. The advantage of deep learning is in acquiring self-organizing features from gathered data; however, it is computationally impractical to implement in real-time interaction applications. Moreover, it can be difficult to attach a physical interpretation. An alternative suggested framework in such cases is integrated perception-action.

In this dissertation, we propose two integrated interactive perception-action algorithms for real-time automated grasping of novel objects using pure tactile sensing. While visual sensing and processing is necessary for gross reaching movements, it can slow down the grasping process if it is the only sensing modality utilized. To overcome this issue, humans primarily utilize tactile perception once the hand is in contact with the object. Inspired by this, we first propose an algorithm to define similar ability for a robot by formulating the required grasping steps.

Next, we develop the algorithm to achieve force closure constraint via suggesting a human-like behavior for the robot to interactively identify the object. During this process, the robot adjusts the hand through an interactive exploration of the object's local surface normal vector. After the robot finds the surface normal vector, it then tries to find the object edges to have a graspable final rendezvous with the object. Such achievement is very important in order to find the objects edges for rectangular objects before fully grasping the object. We implement the proposed approaches on an assistive robot to demonstrate the performance of interactive perception-action strategies to accomplish grasping task in an automatic manner.

To my parents and whomever taught me something!

ACKNOWLEDGMENTS

I gratefully thank my advisor and all members of my committee for their time and valuable comments.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

In the world of active agents, an agent desires to adjust its behavior during interaction with its environment. Such an interaction consists of three main components: sensation, perception, and action. Depending on the task and the sensed modality, these three elements have to be implemented in coordination with each other to have the best performance. Perception is an intermediate step and is responsible for extracting useful information from the sensed data. The quality of extracted information from the sensed data is dependent on the power of designed perception algorithm to mine the data effectively. On the other hand, more powerful algorithms can consume a lot of time and energy from a light robot either in training phase or in inference phase. A promising solution to address these issues is integrated interactive perception approach. In such an approach, the main attention is in minimizing the perception computational processing by coordination of the three components of sensation, perception, and action. Such a coordination highlights the topic of interactive perception and is referred as a learned policy in the context of reinforcement learning. As it is well known, learning this interactive perception from gathered experiments is computationally expensive for light robots; moreover, the learned policy can not be interpreted easily.

In this work, we aim to propose an integrated interactive perception with use of force-torque sensing for grasping task. Grasping task can be initiated by either visual or tactile perception. Although the first one is powerful in terms of object recognition, a continuous processing of visual data during interaction is not a simple task for a light robot. In real world, a human also barely utilizes vision ability in close proximity of the object to be grasped; indeed, tactile perception ability is more useful in the vicinity of object.

A robot can perceive a touched object either though its joints or its wrist. One of the main issue with robot joint data is accuracy in smaller range of sensation. Tactile interaction is dependent on very

small range of sensation accuracy for perception of the object. Larger range of sensing requires to sacrifice other useful properties of the robot behaviour during interaction. For instance, to perceive the object from the sensed joint data, more force should be applied to the object to project the tactile sensing into the robot joints due to effect of accumulated noise in several sensors on the interaction-related data [1]. A more convenient way to perceive an object with light touch is via the sensation in the robot wrist. We use this implementation to interactively explore and perceive the object surface more accurately. As a result, grasping a broader range of objects specifically rectangular objects would be possible.

In this dissertation, we pursue following objectives and contributions: 1) We show that it is feasible to propose a more human-like grasping using tactile touch sensing. 2) We describe an interactive perception to simultaneously identify and grasp an unknown object. 3) We present the procedure to make the interaction between the robot and object compliant. 4) We propose an approach by minimizing sequential logical rules. 5) We develop the algorithm with enough parameters to give freedom in adjustment, maneuvering and interaction during grasping unknown objects. 6) We design the algorithm with reducing the dependency between these freedom of actions. 7) We attempt to generalize the proposed algorithm to broader range of graspable objects without using prior knowledge of object model.

The remainder of this dissertation is organized as follows. Chapter 2 describes related works. In Chapter 3 we describe the mathematical framework of perception-action cycle in control manner using tactile sensing with an algorithm for grasping with experimental results. In Chapter 4 we describe the strategy to interactively perceive and grasp a touched object with capability to satisfy force closure constraint for grasp rectangular objects as well as the relevant demonstrations. We conclude and talk about the future work in Chapter 5.

CHAPTER 2: RELATED WORK

Grasping is an essential and complex daily activity. Through this task, humans show an intention to affect surrounding environment in a controllable manner. Humans primarily utilize a combination of control strategy and learning from repetitive experiments to anticipate grasping in different situations in which the properties of an object such as size, shape, and contact surface are important parameters during grasping task [2]. There is a lot of attention on vision-based grasping during recent years. For instance, reconstruction a 3D geometry mapping order to estimate force using Recurrent Neural Network (RNN) was presented in [3, 4]. Authors in [5] suggested stereo vision-based sensor to estimate the contact force for surgical applications.

Data-driven approaches have emerged during recent year as the machine learning approaches are dominated. A data-driven approach of grasping with vision is proposed in [6]. Data-driven methods have different forms. Human-supervised methods predict grasp configurations [7, 8, 9, 10]. Other methods such as [11] predict finger placement offline from geometry. Deep learning has incorporated into such data-driven approaches [12, 13, 14, 15]. Feedback has been employed into grasping to achieve the desired forces for force closure constraint and other dynamic grasping criteria in [16]. Training a network to predict the optimal grasp from a given image through self-supervised data collection is proposed in [17]. Using a dataset of 3D models, Dex-Net is combined with a learning approach to satisfy force closure constraint in [18]. Data-driven approaches are based on huge dataset of grasping attempts. Parallelizing data collection as well as cloud robotics are the proposed solution in some research works [19, 20, 21, 20]. Some successful grasping methods use synthetic depth-images from large-scale simulated data [22, 12, 8]. Detecting grasp affordance using these depth images suggested in [10] and to plan and execute a grasp [12].

Robotic reaching is related to our work and an important area in robot grasping dealing with

coordination and feedback for reaching motions [23, 24]. Using visual feedback, moving a camera or end-effector to a desired pose is the subject of visual servoing [25] and usually relies on manually designing or specifying the features for feedback control [26, 27]. Prior calibration between the robot and camera is a burdensome situation in visual servoing which has been addressed in [28, 25]. Recent progress in learning and computer vision techniques have also employed in visual servoing [29, 30].

Tactile sensation is a very informative feature to recognize object properties. In [31], the authors proposed a tactile perception strategy to measure tactile features for mobile robots. Tactile sensing also is used to propose a robust controller for reliable grasping [32] and slipping avoidance [33]. Visual sensing and tactile sensing are complementary in robot grasping. A combination of both of them through deep architecture is a promising solution in [34, 35]. However, processing these high-dimensional data is not an easy task and a meaningful compact representation would be needed [36, 37]. A robot can learn the manipulation using tactile sensation through demonstrations [38, 36, 39].

A lower dimensional representation of tactile data is also more useful for object material classification [40]. With more processing approaches such as bag-of-words, identification of objects would be possible in advance [41]. Extraction of object pose via touch based perception can be used for manipulation [42]. Moreover, localization will be improved by contact information gathered by tactile sensor [43, 44]. The robot can control and adjust the pose of hand with stability consideration after evaluating the tactile experiences [45, 46].

Tactile feedback and interactive perception are very important components of the grasping task. An introduction of predictive force control and reactive control strategies in this domain is provided in recent years [47]. Interactive perception has been introduced as a potential field of study in recent years emerging from the fact that the perception can be facilitated by interaction with the

environment [48].

A couple of works [45, 49, 46] suggested tactile sensors to estimate grasp stability. By adapting the grasp, [50] integrated tactile sensing with objects dynamics models to adapt the grasping for a dexterous hand. In [51, 52, 53] the tactile signals used to mine the features in order to predict slip and adaptively adjust the grasping force. A combination of visual and tactile information using model-based methods is presented in [54, 55, 56, 57, 58]. Improvement over single-modality sensing is reported in these works. Nevertheless, such approaches are based on accurate robot and object models as well as tactile sensors calibration. Another similar work is presented in [59] in which a re-grasping policy by tactile sensing via a learned stability metric proposed. A heuristic transition function utilized to predict future tactile data.

End-to-end learning is another attractive approach during recent years in which there is no need to any prior model or transition function [60, 61]. Such approaches are based on learning action-conditioned model from gathered data. Authors in [61] proposed a tactile re-grasping approach using the GelSight sensor simulating transformations to tactile data based on rigid body dynamics. Focusing on tactile localization without vision [62] explores grasping with a 3-axis force sensor.

Touch-based control approaches employ manual design of control commands to extract high-level touch sensors features [63, 64]. Using a high resolution touch sensor, learning general-purpose predictive model is the proposed solution to overcome the efforts to prespecify the control laws and features [65]. Reinforcement learning can be used to learn stabilizing an object with touch sensing [36].

Grasp planning is an important phase of robot manipulation and has been attractive for the robotics research community from past decade [66, 67, 68, 69, 70]. Recently, model based using a grasp quality metric based on local properties of the camera point cloud to identity the object [71, 12]. To localize grasp points from the RGBD image deep learning and computer vision techniques are

in the center of attention in [15, 17, 72]. While these approaches rely on open-loop frameworks, recent attempts are in the direction of learning closed-loop control policy via deep reinforcement learning [6]. Combining RGBD cameras with infrared sensors, enhancement in the robustness and adaptability of grasping an unknown objects with uncertain position is possible by designing a reactive algorithm [73].

Specifically, researchers have turned to tactile sensing to enable more effective closed-loop approaches to robotic grasping based on robot feeling rather than visual inspection [74]. Tactile sensors are very effective during different phases of grasping such as detecting the contact slip [75, 76, 77], estimating the contact forces [78] as well as localizing objects [79, 80]. Tactile feedback is utilized in reinforcement learning approach alongside grasp quality predictor to learn grasp adjustments [39]. Tactile sensors are capable of integrating into dynamic motion primitive to enable associative skill memories [81].

CHAPTER 3: GRASPING USING TACTILE SENSING

To implement our proposed approach on automatic grasping based on tactile sensing, we first describe the mathematical framework of perception-action cycle during interaction with an object using tactile sensing.

The Six-axis Force/Torque Tactile Data

Grasping is a physical interaction with environment. During such interaction, the exchanged data between the object and robot would be the force and torque data. The robot experiences torque data τ_f as a consequence of inserted force f as shown in Figure 3.1. In this figure, three directions for the sensed force are marked to be recognizable. The direction of this force (①, ② and ③) is dependent on the direction of robot hand movement toward the object v and it is in opposite direction of movement after contact with the object surface:

$$sign(f) = -sign(v) \tag{3.1}$$

where the sign function returns the sign of the data. The sensed force vector can be decomposed into the robot hand frame x, y and z in Figure 3.1. So, the 3-axes of tactile force data are the set of $f_e = \{f_x, f_y, f_y\}$.

The remaining part of tactile sensing is the torque data. From Figure 3.1, the resulting sensed torque τ_f tends to rotate the hand in the clockwise direction. This direction would be valid for all the marked force directions in the figure and since it is actually around the x coordinate:

$$\tau_x = \tau_f \tag{3.2}$$

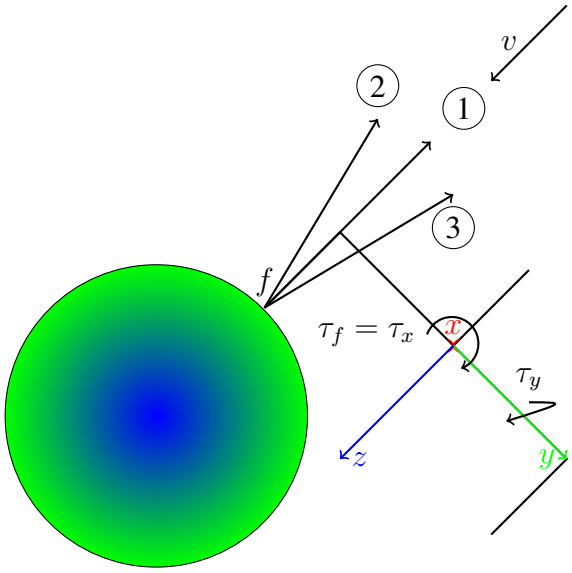

Figure 3.1: Robot hand interaction with object and force/torque tactile sensing

This generated torque can be expressed by cross product of the lever arm vector and the sensed force vector as follow:

$$\tau_x = \vec{r} \times \vec{f} = ||r||||f||sin\theta \tag{3.3}$$

where θ is the angle between the lever arm vector and the force vector. In three dimensional space, the interaction force f can also result in torque around y direction denoted by τ_y. The last part of torque tactile sensing set is τ_z. Any movement in x and y coordinates during contact with object would generate a torque around z axis. This torque is the consequence of friction force during movement on the object surface and the friction force is dependent on the magnitude of normal force inserted on the surface f. If we assume a constant friction coefficient μ for surface we can expand (3.3) as:

$$\tau_z = \vec{r} \times \vec{f} = \mu||r||||f||sin\theta \tag{3.4}$$

As a consequence, the torque tactile data consists of the set of $\tau_e = \{\tau_x, \tau_y, \tau_z\}$. All the six-axis force/torque tactile data set is essential for grasping task. These set of tactile data F_e are useful to

guide the robot hand during interaction with the object surface:

$$F_e = \begin{bmatrix} f_e \\ \tau_e \end{bmatrix} \tag{3.5}$$

Interactive Perception-Action Cycle Through Feedback Control Command

As the tactile data is available, we can present the procedure to define the grasping steps for a robot using the six-axis tactile data. We can use the interaction force/torque data to guide the robot hand in the vicinity of the object. This data is informative enough to define the required grasping steps through an interactive perception-action cycle. As the robot perceives the object via tactile sensing it can generate the suitable action in response to its perception in a cyclic manner shown in Figure 3.2.

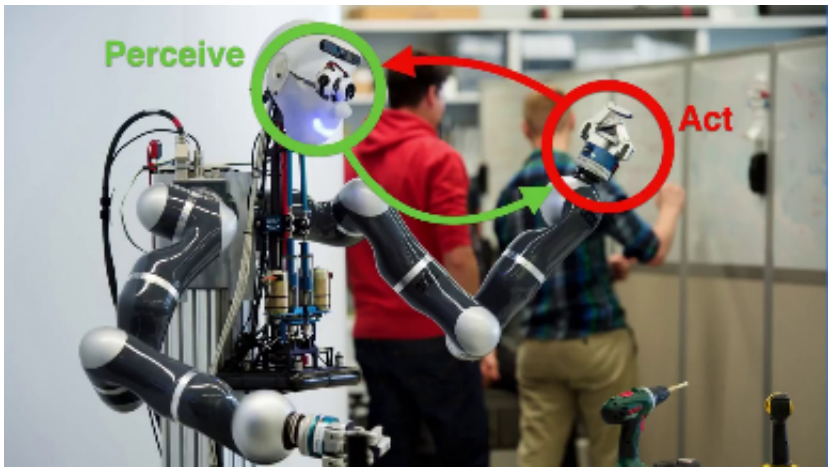

Figure 3.2: Interactive perception-action cycle[1]

[1] https://am.is.tuebingen.mpg.de/uploads/publication/image/18783/LoopLearning.png

Our approach is based on feedback tactile sensing to adjust the robot hand velocity in corresponding direction during its interaction with the object as it is shown in Figure 3.1. With this goal, we assume dynamics for linear velocity v and angular velocity ω of robot hand as follows:

$$a_v \dot{v} + v = u_v \tag{3.6}$$

$$a_\omega \dot{\omega} + \omega = u_\omega \tag{3.7}$$

Where u_v is the command to adjust the linear velocity and u_ω to adjust the angular velocity. The constant coefficients a_v and a_ω are the system inertia to damping ratio and determine the time profile of linear and angular velocity in response to the adjustment command respectively.

According to Figure 3.1, any change in hand location around the object surface will be affected by commanding the linear velocity. Meanwhile, any rotation around the object to point the hand fingers toward the object can be accomplished via influencing the angular velocity. As the robot fingers are interacting with the object surface, the adjustment in the commands is needed to change the hand location or the angle between the hand and the object. The control command for linear velocity is defined as:

$$u_v = \alpha_v (f_f - f_e) \tag{3.8}$$

Where f_e can be any element of force tactile sensing set $f_e = \{f_x, f_y, f_z\}$ and α_v is a necessary scaling constant to scale two different domains (force in several $Newton$ and velocity typically in a few mm/s). The desired value f_f is the desired final value that we consider for that direction. For instance, if the robot needs to touch the object, it has to move directly toward the object according to Figure 3.1. In such case, the control command has to be designed for linear velocity in hand z direction with $\alpha_{vz} < 1$:

$$u_{vz} = \alpha_{vz} (f_{zf} - f_z) \tag{3.9}$$

Executing (3.6) with this command, the robot hand moves with a constant velocity v_{dz} given as

$$v_{dz} = \alpha_{vz} f_{zf} \tag{3.10}$$

toward the object until the force tactile sensing f_z converges to f_{zf}. Nevertheless, the velocity of movement is dependent on the final touch force f_{zf}. With a small modification, we replace the control command with following alternative:

$$u_{vz} = v_{dz}(1 + \frac{f_z}{f_{dz}}) \tag{3.11}$$

where v_{dz} can be selected freely and the final contact force is adjusted by f_{dz} as follows:

$$f_{zf} = -f_{dz} \tag{3.12}$$

A command in form 3.11 make the interaction between the object and the robot hand compliant. This is an useful property for a robot agent making it adaptable during its interaction with other agents or an external environment [82, 83, 84].

Similarly, to rotate the hand around each axis, the corresponding control command can be defined by a scaling coefficient

$$u_{\omega} = \alpha_{\omega}(\tau_d - \tau_e) \tag{3.13}$$

where $\tau_e = \{\tau_x, \tau_y, \tau_z\}$ is the feedback torque tactile sensing. As another example, if we need to align the hand by rotating the hand around its z axis, we can consider:

$$u_{\omega z} = \alpha_{\omega z}(\tau_{dz} - \tau_z) \tag{3.14}$$

Inserting this command into (3.7) after a transient time imposed by the system inertia to damping

ratio, the hand rotates with a constant angular velocity ω_z equals

$$\omega_z = \frac{\alpha_{\omega z}\tau_{dz}}{b_{\omega z}} \tag{3.15}$$

which will continue till the corresponding tactile sensing measurement τ_z reaches the level of desired torque τ_{dz}. Similar to 3.11, it is possible to decouple the rotation velocity and the corresponding sensed torque by:

$$u_{\omega z} = \omega_{dz}(1 + \frac{\tau_z}{\tau_{dz}}) \tag{3.16}$$

In which the final sensed torque would be:

$$\tau_z = -\tau_{dz} \tag{3.17}$$

Now, we can define an intuitive grasping procedure according to Figure 3.1 using the proposed primitive commands as follows:

- Move directly toward the object in z direction
- Stop movement when the fingers touch the object
- Keep a constant inserted force in the z direction
- Rotate around the contact point using torque tactile data

All these defined steps can be executed by both proposed commands in (3.11) and (3.14) with some minor modifications. The three first steps are encapsulated in (3.11). To rotate around the contact point using tactile torque data, we propose the following commands:

$$u_{\omega x} = \alpha_{\omega x}\tau_x \tag{3.18}$$

$$u_{\omega y} = \alpha_{\omega y} \tau_y \tag{3.19}$$

Where $\alpha_{\omega x}$ and $\alpha_{\omega y}$ are the scaling factors for the corresponding directions. Since the sensed torque around y axis is dependent on the small width of finger tip, we can pass that corresponding small torque through a non-linear activation function such as $tanh$ with an argument gain of β_y:

$$u_{\omega y} = \alpha_{\omega y} \tanh(\beta_y \tau_y) \tag{3.20}$$

The flowchart of proposed approach is provided in Figure 3.3. According to this diagram, the robot attempts to reach to zero sensed torques in x and y directions. There are two indicators for alignment in x and y direction called as $S_{\omega x}$ and $S_{\omega y}$. As both of them indicate the alignment with object surface, a switch inside the hand palm makes the decision on the object final situation with respect to the hand. It would be possible that in some rare cases the object be inside the hand simply by moving toward the object and the grasping process will be terminated at the beginning stage. Also, it would be possible that the object is not graspable either the object is bigger than the two fingers width or the algorithm is not generalized enough to grasp the object.

We use a wrist force/torque sensor to extract tactile sensing from the interaction with the object. Nevertheless, the sensor reports the bias information related to the weight of hand alongside with interaction data. In next section, the mathematical model of this bias information is provided.

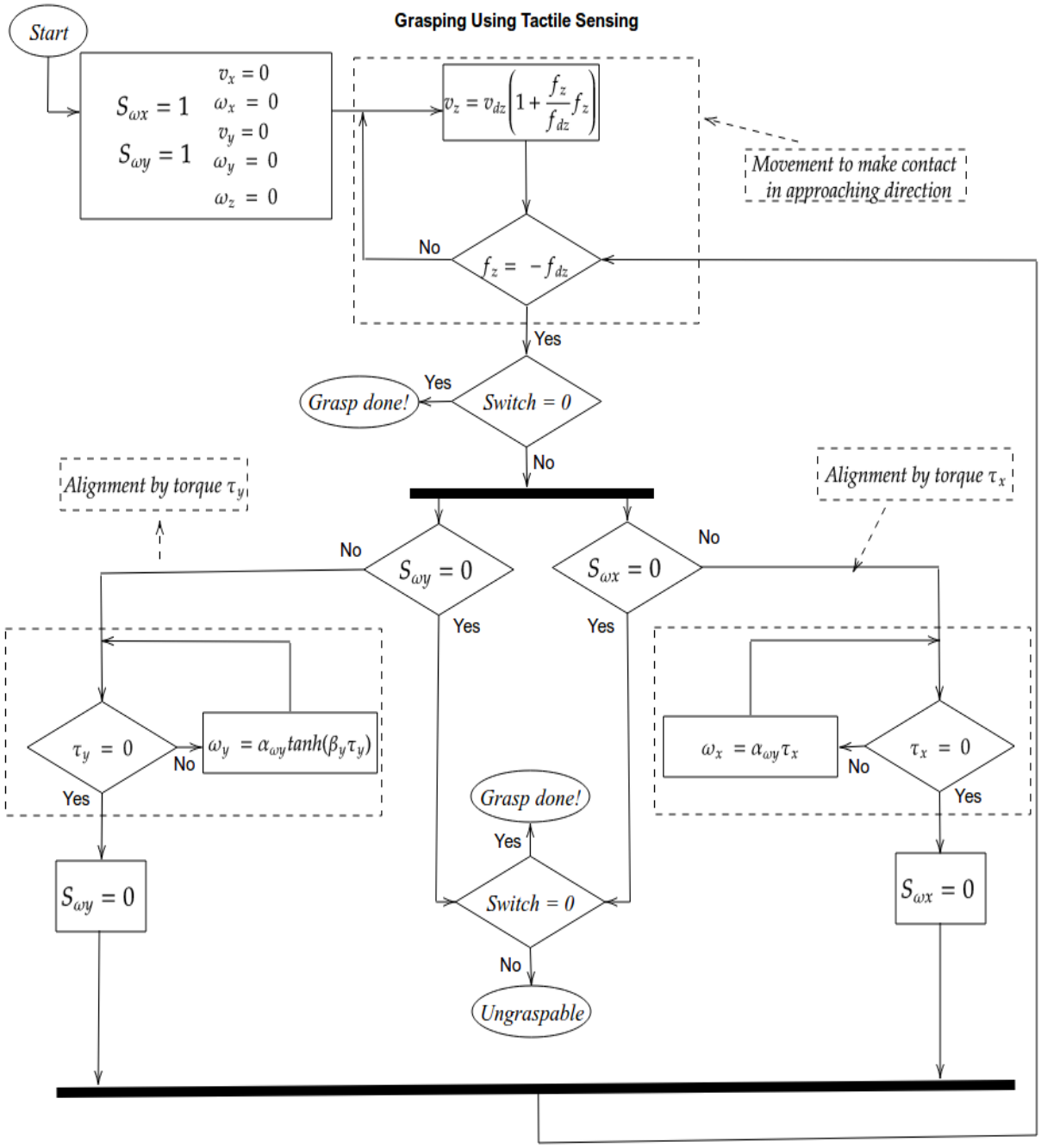

Figure 3.3: The flowchart of grasping algorithm using tactile sensing

Wrist Force Torque Sensor Calibration

The tactile data is collected from the wrist mounted on the robot wrist. The sensor is connected to the robot hand and during the motion this information is also reported with interaction information. This bias data needs to be removed before further processing of remaining data. The bias data is coming from the hand weight and is a result of gravity effect on the mass of hand affecting both force and torque as shown in Figure 3.4.

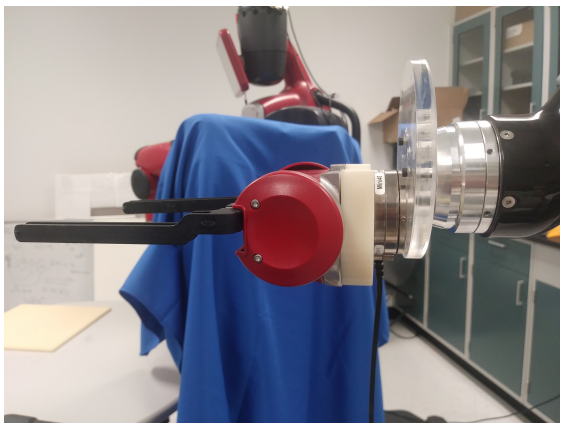

Figure 3.4: Force/torque sensor mounted on the robot wrist

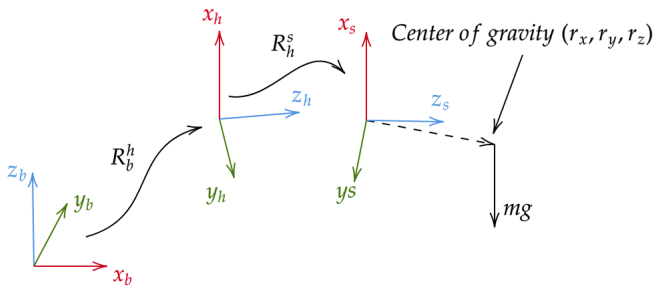

Figure 3.5: Center of gravity and rotation matrix for sensor bias calibration

As the robot hand maneuvering around the object, the center of gravity is changing proportionally with the rotation matrix between the robot hand and the robot base frame as shown in Figure 3.5. According to Figure 3.5, we can find the gravity force effect by projecting the gravity force from base frame to the sensor frame using the rotation matrix between the sensor frame and robot hand frame (R_h^s):

$$F_{bias} = R_h^s R_b^h mg \tag{3.21}$$

Meanwhile, the gravity torque effect is extracted using the coordination of center of gravity $r = (r_x, r_y, r_z)$ expressed in the robot base frame:

$$\tau_{bias} = (R_b^h)^T (R_h^s)^T (R_h^s R_b^h r \times mg) \tag{3.22}$$

In which the cross product $\times$ is using to formulate the relationship between the force and the torque.

In order to identify the coordinates of center of gravity $r = (r_x, r_y, r_z)$, we expand the relationship between the sensor measured torque τ_s and measured force f_s via cross product:

$$r \times f_s = \begin{bmatrix} i & j & k \\ r_x & r_y & r_z \\ f_x & f_y & f_z \end{bmatrix} = (r_y f_z - r_z f_y)i - (r_x f_z - r_z f_x)j + (r_x f_y - r_y f_z)k \tag{3.23}$$

The most right hand side of this relationship is equivalent to:

$$\tau_x = (r_y f_z - r_z f_y), \tau_y = -(r_x f_z - r_z f_x), \tau_z = (r_x f_y - r_y f_z) \tag{3.24}$$

Recording the data from three different configurations, the center of gravity elements are identifiable from 3.24. In other case in Figure 3.4, these elements are:

$$r = [r_x, r_y, r_z] = [-0.0047, -0.001156, 0.0477] \tag{3.25}$$

The interaction data is the subtraction of raw measured data from the bias data in equation 3.21 and equation 3.22. We can investigate the performance of algorithm presented in Figure 3.3 in next section.

Experimental Results

In this section, we present the experimental settings, the time profile of tactile sensing set with demonstrations for a typical grasping of two objects.

Settings

To validate the proposed approach, we use a Mico robot with 6 degrees of freedom equipped with a wrist force/torque sensor and a parallel gripper. We select cylindrical and a rectangular shaped objects with misalignment between the initial configuration of the robot hand and the object surface. We present the sensed interaction force and torque as well as robot hand linear and angular

velocities and roll, pitch and yaw angles during interaction for each demonstration in this context. The convention used for these angles representation can be found in Figure 3.6.

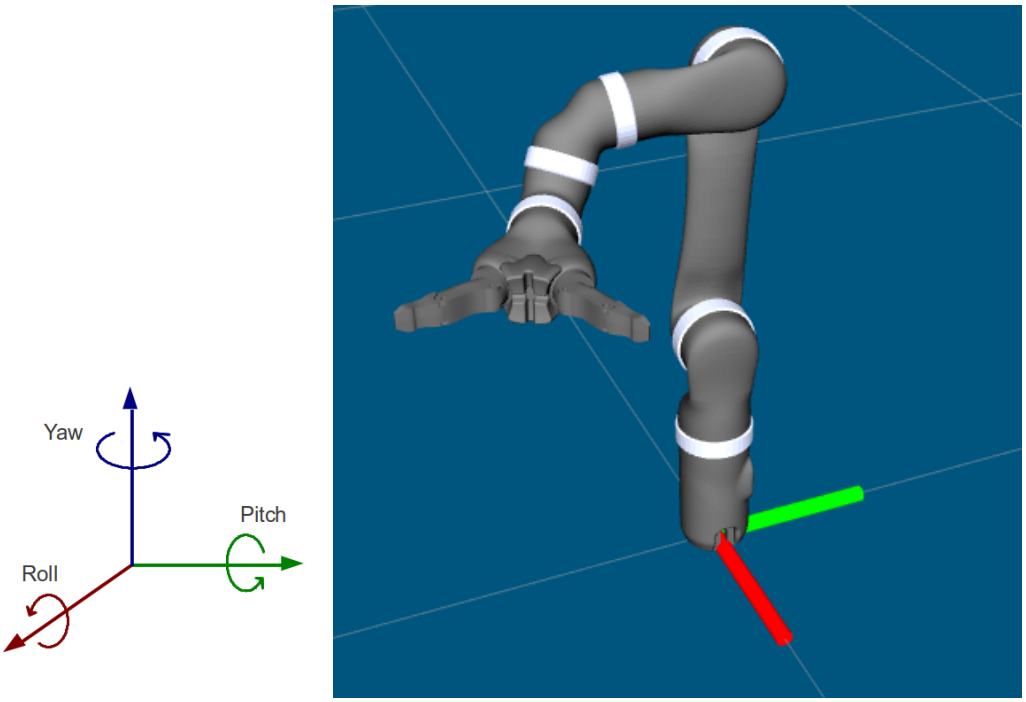

Figure 3.6: Mico robot with hand frame angles representation convention

We consider $f_{dz} = -0.7$ to achieve the corresponding level of touch force and make sure there is enough sensed torque to rotate around the contact point. The parameters used in the experiment and the algorithm flowchart 4.7 are listed in Table 3.1.

Table 3.1: Parameters

Parameter	Value
v_{dz}	0.0055
f_{dz}	-0.7
$\alpha_{\omega x}$	5
$\alpha_{\omega y}$	0.075
β_y	120

Demonstrations

Rectangular object:

The initial configuration of rectangular candidate object is shown in Figure 3.7.

There is both pitch and yaw misalignment angle between the robot hand the object surface. The initial pitch misalignment is 18.43° and the torque around y axis τ_y compensates for this misalignment. Meanwhile, the final yaw angle to grasp the object from an initial yaw angle of 18.4° is -27.53° and the torque around x axis is responsible to reach to this level to successfully grasp the object. The video of the experiment is available at grasping rectangular object. Tactile data during grasping the object is shown in Figures 3.8 and 3.9. Linear and angular velocities can be found in Figures 3.10 and 3.11. The time profile of pitch, roll and yaw angles is represented in Figures 3.12.

Figure 3.7: Initial configuration of grasping rectangular object

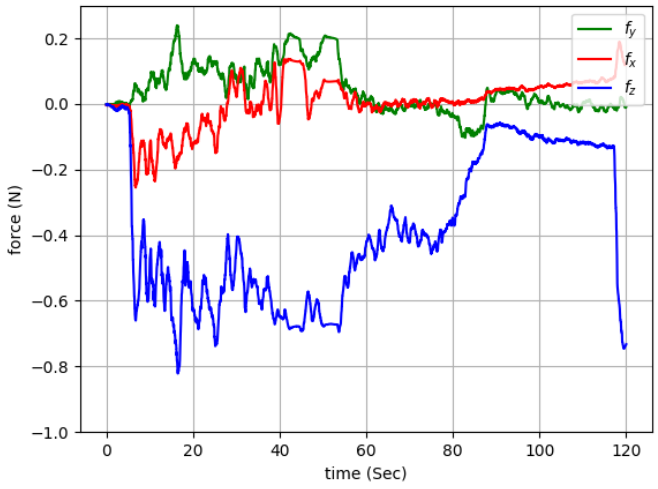

Figure 3.8: Tactile forces

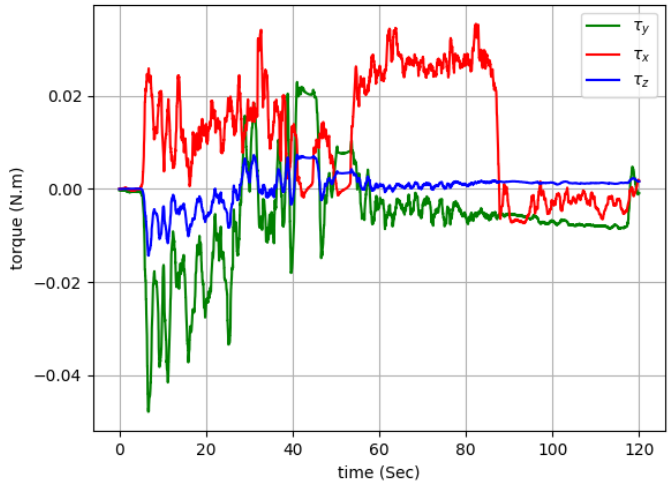

Figure 3.9: Tactile torques

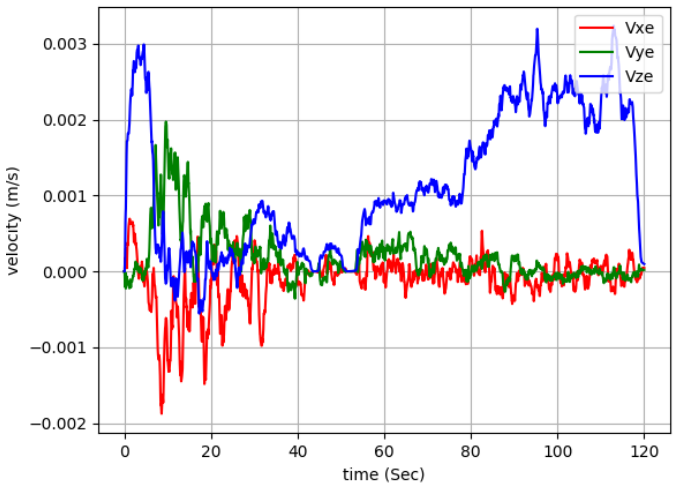

Figure 3.10: Linear velocities

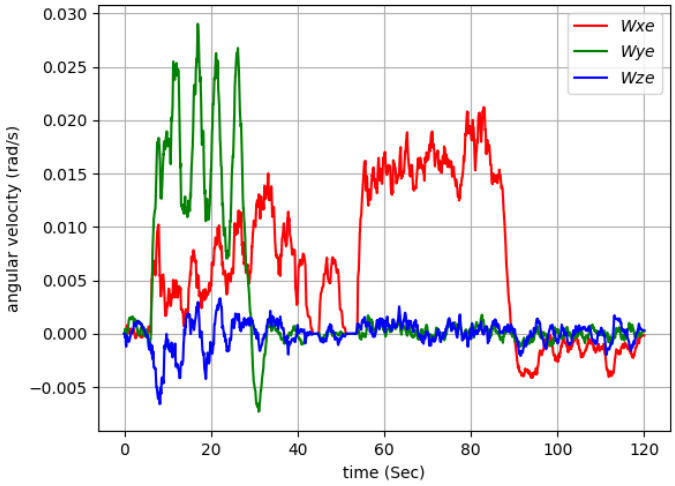

Figure 3.11: Angular velocities

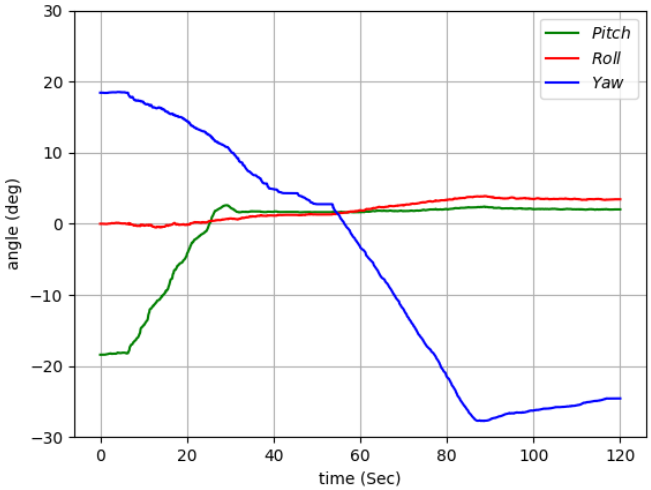

Figure 3.12: Angles

According to Figure 3.12, the pitch angle converges to near zero. However, the yaw angle decreases

over time to finally align the hand with object surface. Since the algorithm uses the sensed torque to align with object surface, it needs more rotation effort in x through changing the yaw angle to grasp the object as is shown in Figure 3.13. Moreover, the touched force converges to our desired value in Figure 3.8.

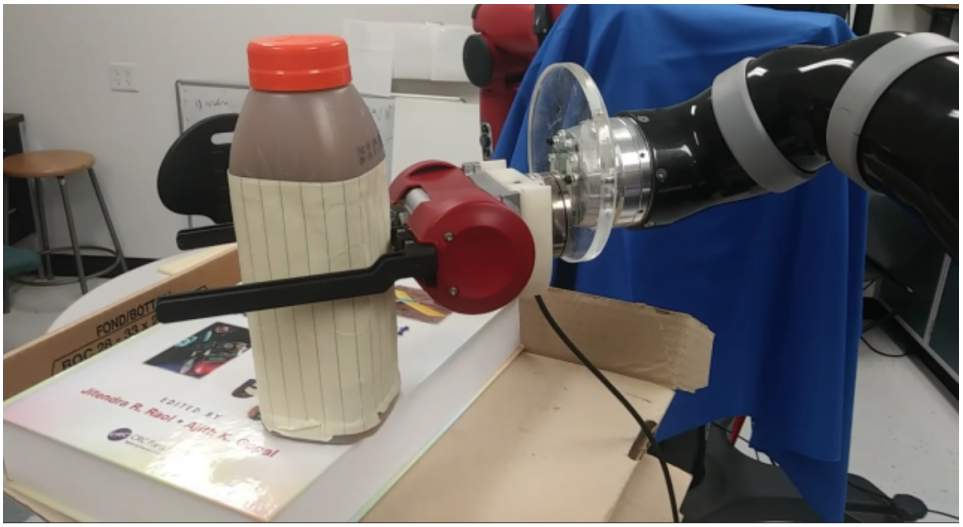

Figure 3.13: Final configuration of grasping rectangular object

Cylindrical object:

The initial configuration of candidate object for cylindrical object is shown in Figure 3.14.

There is both pitch and yaw misalignment angle between the robot hand the object surface. The initial pitch misalignment is -18.43° and the torque around y axis τ_y compensates for this misalignment. Meanwhile, the final yaw angle to grasp the object from an initial yaw angle of 18.4° is -15.68° and the torque around x axis is responsible to reach to this level to successfully grasp the object. The video of the experiment is available at grasping cylindrical object. Tactile data during grasping the object is shown in Figures 3.15 and 3.16. Linear and angular velocities can be found

Figure 3.14: Initial configuration of grasping cylindrical object

in Figures 3.17 and 3.18. The time profile of pitch, roll and yaw angles is represented in Figures 3.19.

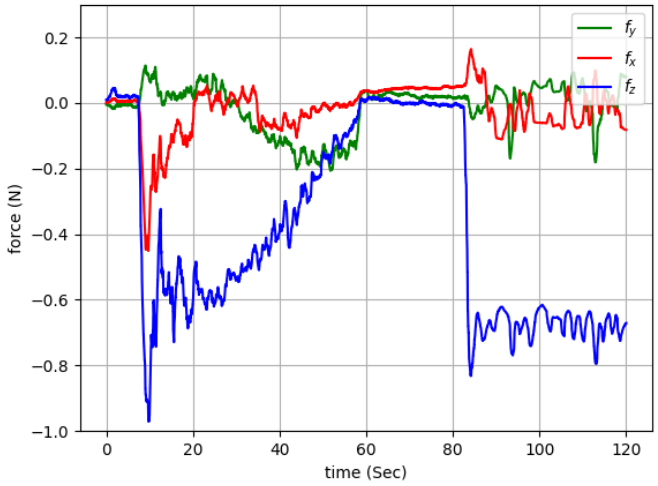

Figure 3.15: Tactile forces

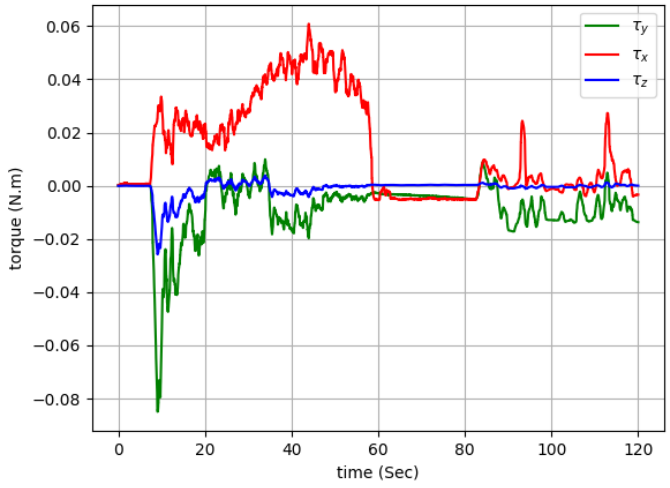

Figure 3.16: Tactile torques

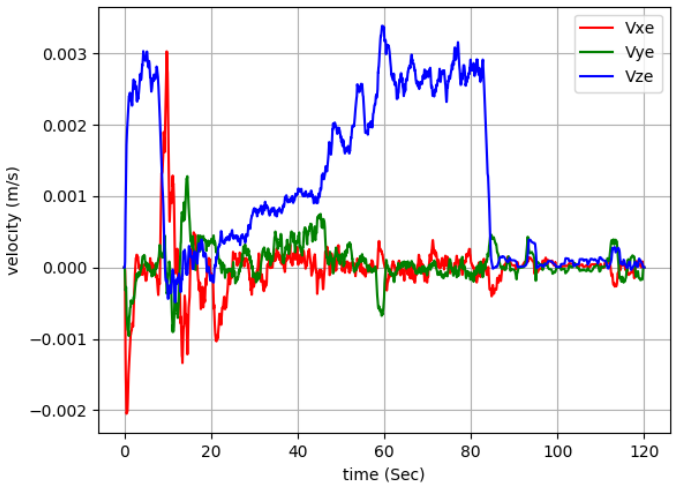

Figure 3.17: Linear velocities

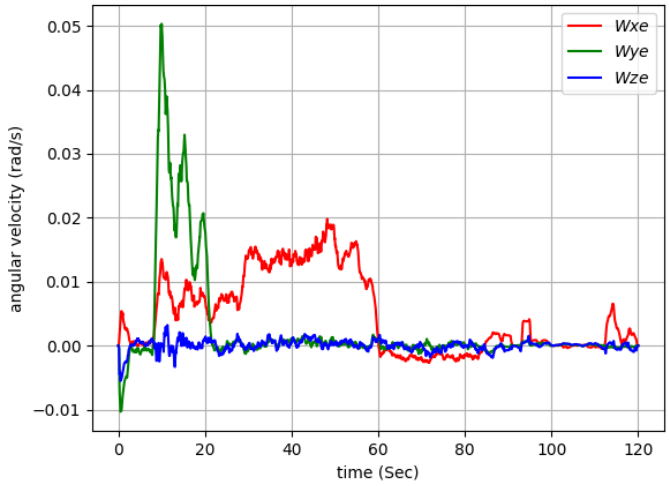

Figure 3.18: Angular velocities

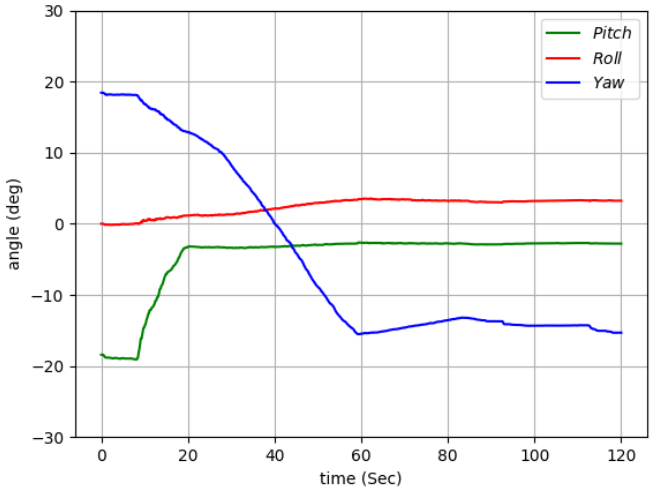

Figure 3.19: Angles

In this case, grasping of cylindrical object is faster than rectangular object. This fact can be inferred

from the final yaw angle in Figure 3.12 and Figure 3.19. Because the robot finds the edge to grasp the object with less effort in comparison with the rectangular object. The final situation of object grasping is shown in Figure 3.20.

Figure 3.20: Final configuration of grasping cylindrical object

The proposed algorithm has limitation with a broader range of graspable rectangular shape objects. This limitation is highlighted in chapter 4 with suggested enhancement in the algorithm to generalize the algorithm to more graspable situations.

CHAPTER 4: GRASPING USING TACTILE SENSING WITH FORCE CLOSURE CONSTRAINT

In this chapter we provide the approach to grasp an object considering the force closure constraint based on the modification in the proposed approach in chapter 3. Force closure constraint means any possible motion of the object is resisted by the contact force. It is an important part of grasp planning and necessary condition to grasp the object successfully [85].

Limitation of Previous Algorithm

There are two assumptions with the previous algorithm. The first one is that the touched force is enough to generate the required torque to rotate and grasp the object. As a consequence of this assumption, there is a dependency between the touched force and the required sensed torque. The second assumption with the previous algorithm is that the object will be inside the robot fingers after rotation around the contact point according to sensed torque data. Although this is a reasonable assumption for a set of graspable cylindrical and small rectangular object, it would not be the case for special case of rectangular objects with limited range of grasping in finger width as is shown in Figure 4.1. In such case, force closure constraint is not satisfied.

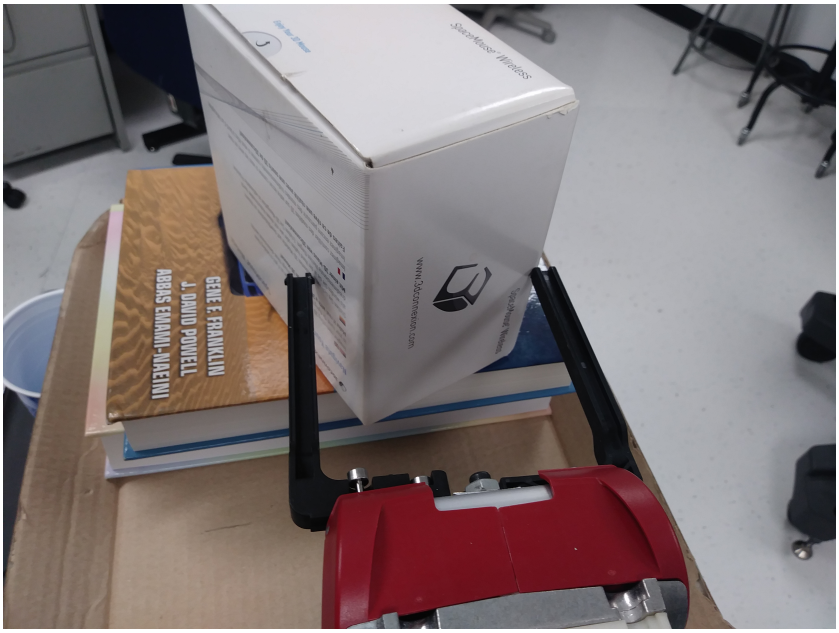

Figure 4.1: Ungraspable situation using torque for rotation

To address these two limitations, we modify to improve our proposed algorithm. Our solution is to simultaneously identify and adjust the robot hand to the surface normal vector and then try to grasp the object. According to the Figure 4.1, the object is not inside the robot fingers. Once the fingers align with the normal vector of front view surface, grasping is feasible as shown in Figure 4.2.

Normal vector is an indication of surface curvature pointing out of the object surface. It is orthogonal to the surface tangential plane on the surface illustrated in Figure 4.3. According to this Figure, the normal vector can be used to align the fingers with the object surface.

Inspired by humans in similar situation, we define same behaviour for the robot to identify the normal vector. In this approach the robot interactively explores the local contact point based on its feeling during exploration and simultaneously adjusting the hand accordingly. According to

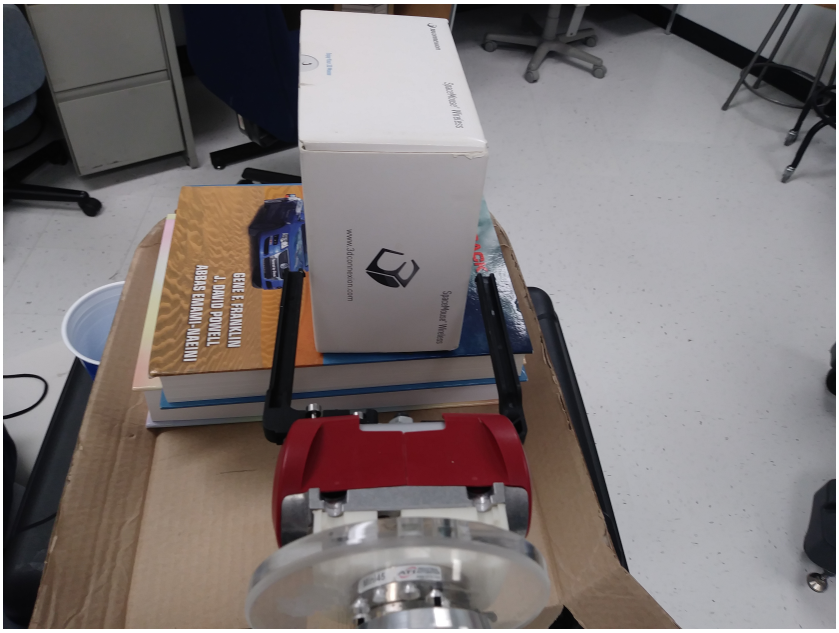

Figure 4.2: Graspable situation with alignment to the surface

Figure 4.3, the action to extract such information is a linear back and forth movement v_y in the direction of hand width (y direction) around the contact point.

Such movement results in unbalance sensed force in the direction of movement. Figure 4.4 presents the decomposition of interaction force during hand movement on the object surface. According to this decomposition, robot movement as a consequence of a velocity command in either $+v_y$ and $-v_y$ directions generates an opposite friction sensed force. Nevertheless, an extra reaction force in y direction can amplify the f_y because of misalignment angle θ. Generating a back and forth movement for the robot hand excites this useful opposite reaction force to feel the misalignment between the finger and the object surface.

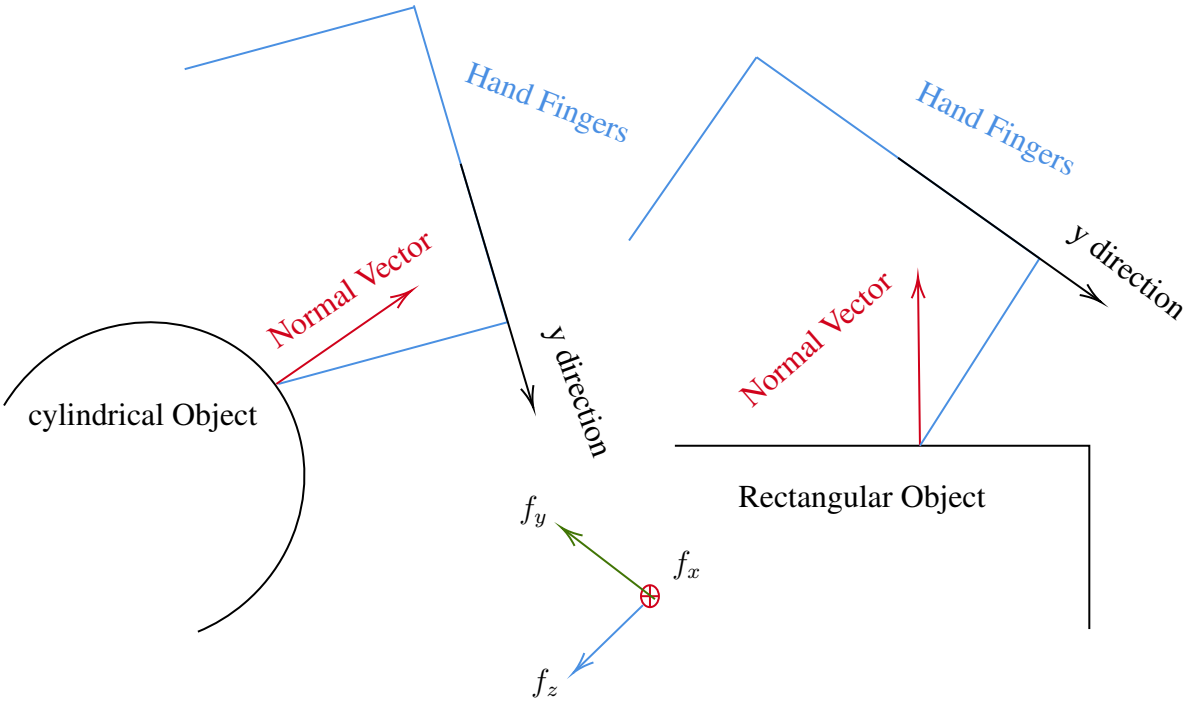

Figure 4.3: Normal vector and hand fingers

The reaction force will follow the changes in command velocity with a latency in measurement and system response to command. The proposed algorithm is to make decision as a result of this sensed force. The decision making process adjusts the hand in a control manner getting feedback of the average of reaction force for a period of movement on the surface. Since the average sensed force is important in our algorithm, we refer to this average force as a **feeling force** in this context.

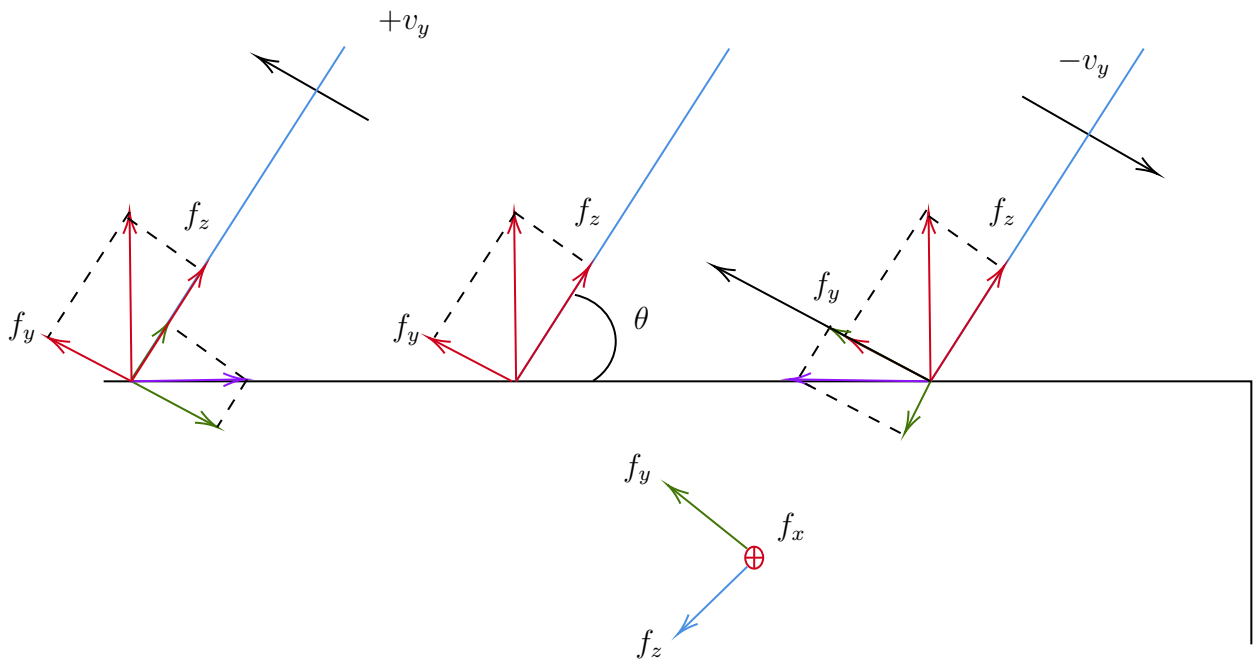

Figure 4.4: Interaction force decomposition during movement

The feeling force provides a perception of surface and the robot can identify the relative situation of its hand with the surface through this perception.

As the robot feels the surface, it can try to align with the object surface while keeping contact with it. Alignment with object surface corresponds to detecting the surface normal vector as a feature of object curvature. To detect this normal vector, we define a human-like behavior for the robot agent. To do so, we need to consider an active surface exploration to find the object curvature. In this regards, we transfer a human-like behavior to the robot by continuously sensing the surface curvature during interaction with object. This behavior can be generated via periodic movement in

y to the tangential plane of surface direction as shown in Figure 4.5 after contact with the object:

$$u_{vy} = v_{dy} sign(sin(\Omega t)) \tag{4.1}$$

In which $\Omega = \dfrac{2\pi}{T}$ is the frequency of feeling with a period of T.

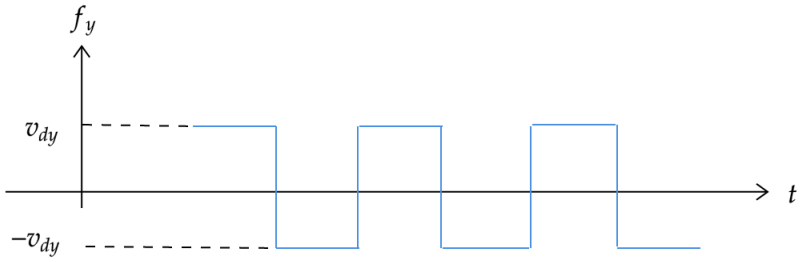

Figure 4.5: Square velocity generation for feeling the object surface

Corresponding to this periodic movement, the robot feels a periodic force in y direction. The magnitude of this feeling force in each positive and negative y direction depends on the misalignment angle with the surface as well as surface friction force. The level of feeling force in each positive and negative directions will change by rotating around the contact point. To align with the surface during interactive perception, this feeling force f_f will be used as a feedback to command the angular velocity in equation 3.18 exponentially decreasing over time:

$$u_{\omega x} = \alpha_{\omega x} f_f \tag{4.2}$$

Alignment situation is the condition in which the feeling force reaches to a certain level. A typical profile of feeling force is shown in Figure 4.6.

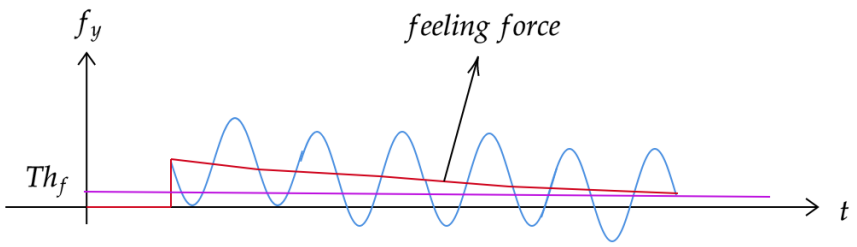

Figure 4.6: Feeling force profile

With these modifications in algorithm, a flowchart of algorithm is presented in Figure 4.7. According to this flowchart, the grasping process starts with initialization in linear and angular velocity command. The robot moves toward the object surface in its approaching direction in a constant velocity $v_z = v_{dz}$. The grasping will be terminated in case that the sensed force in approaching direction reaches to final value and the robot stops of further movement in approaching direction and the switch indicates that the object is inside the fingers. Otherwise, the robot aligns its hand with the object surface monitoring the sensed torques τ_x and τ_y simultaneously and using the feedback information. Alignment with the object surface in the direction of fingers is via applying command pulse velocity and monitoring the feeling force feedback. The indicators $S_{\omega x}$ and $S_{\omega y}$ show alignment and in case the object is not inside the hand, the robot can try to move with $v_y = -v_d tanh(\beta\tau_x)$ to the object edges. $Tanh$ function helps to reflect smoother sensed torque in the command velocity. The reason of using τ_x is that this information provides information about the contact finger with the object. Movement in the direction of this contact finger will result in finding the object edge and grasp it. Otherwise, such movement results in flipping in the sensed torque detected by $\tau_x S_{\tau_x}$. Further rotation around the wrist is required to grasp the object if such case arises. The direction of rotation can be specified via $sign(\beta_z\tau_z)$ and if the rotation is more than 180 degree and the switch could not detect any contact, the object is not graspable.

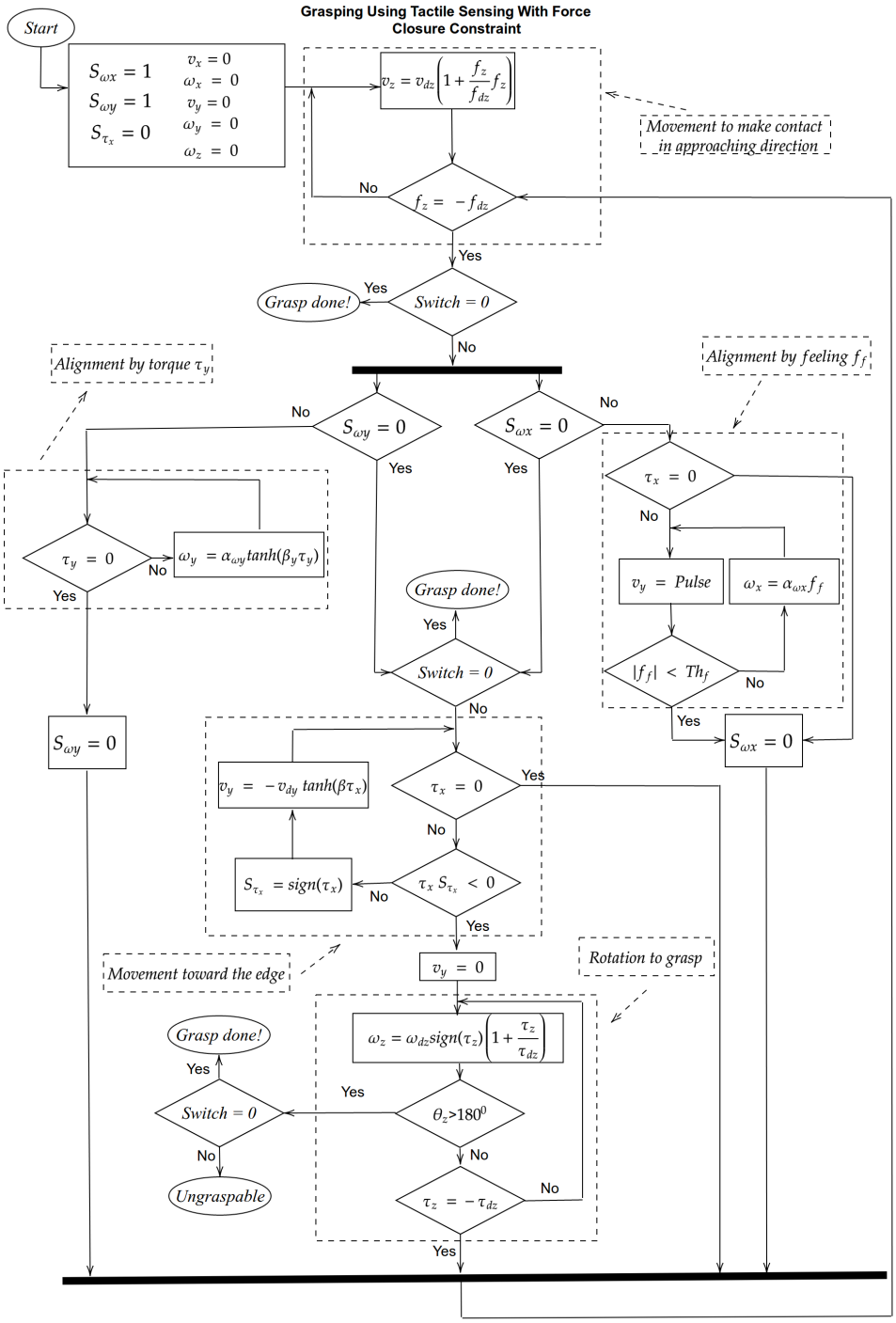

Figure 4.7: The flowchart of grasping algorithm using tactile sensing with force closure constraint

Mathematical Description

Grasping is the process of approaching, alignment and seizing an object in front of hand. The alignment phase is the most important part and we define the object frame and hand frame to describe misalignment situation in Figure 4.8.

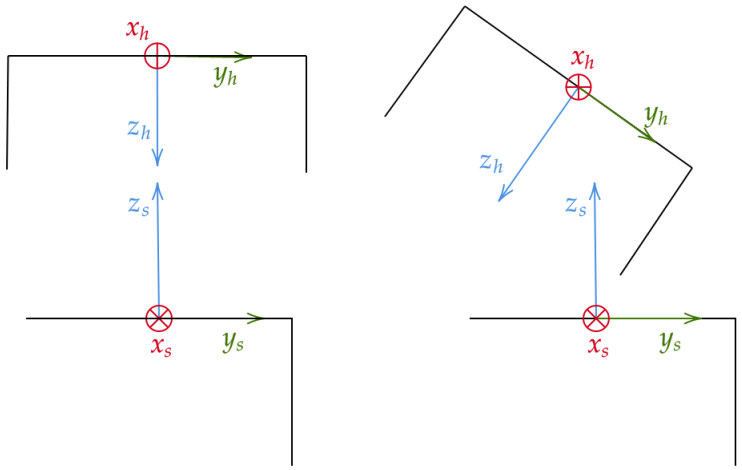

Figure 4.8: Misalignment between the robot hand and the object

The reaction force from the object surface is described as:

$$F_s = \begin{bmatrix} -F_{xs} sign(v_{xs}) \\ -F_{ys} sign(v_{ys}) \\ F_{zs} \end{bmatrix} \tag{4.3}$$

In which $F_{zs} = K_s \Delta z_s$ in which K_s is the springiness coefficient and Δz_s is the deflection in z_s direction. F_{xs} and F_{ys} are the friction forces in the opposite direction of movement.

Assuming misalignment between the object and the robot hand via an unknown rotation matrix R_s^h, we can transfer the above object frame force into the robot hand frame F_h:

$$F_h = \begin{bmatrix} F_{xh} \\ F_{yh} \\ F_{zh} \end{bmatrix} = R_s^h F_s \tag{4.4}$$

A complete grasping is corresponding to alignment with the object surface. In alignment situation, the robot hand frame coordinates $\{xh, yh, zh\}$ has to be in the direction of surface frame coordinates $\{xs, ys, zs\}$. From Figure 4.8 after alignment the rotation matrix R_s^h is:

$$R_s^h = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{4.5}$$

To reach to this final situation, the robot commanded to insert a constant force in hand frame approaching direction. At the same time, the rotation matrix will be influenced via the angular velocities in corresponding direction. To make it clear, we decompose the rotation matrix into the multiplication of three basic rotation matrices:

$$R_s^h = R_s^h(\alpha) R_s^h(\beta) R_s^h(\gamma) \tag{4.6}$$

In which α, β and γ represent yaw, pitch and roll angles, respectively. Each of these rotation matrix

is a basic rotation matrix:

$$R_s^h(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\alpha) & -sin(\alpha) \\ 0 & sin(\alpha) & cos(\alpha) \end{bmatrix} \tag{4.7}$$

$$R_s^h(\beta) = \begin{bmatrix} cos(\beta) & 0 & sin(\beta) \\ 0 & 1 & 0 \\ -sin(\beta) & 0 & cos(\beta) \end{bmatrix} \tag{4.8}$$

$$R_s^h(\gamma) = \begin{bmatrix} cos(\gamma) & -sin(\gamma) & 0 \\ sin(\gamma) & cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.9}$$

The final rotation matrix of multiplication of these three rotation matrices would be as equation 4.5 if we have:

$$\alpha = \pi, \beta = 0, \gamma = \pi \tag{4.10}$$

The above condition is achievable as long as we find the surface normal vector through the measurements from the sensor attached to the robot hand. The flowchart of proposed algorithm in Figure 4.7 presents the procedure to find this normal vector and alignment situation through alignment by feeling and alignment by torque highlighted blocks. As the robot finds the alignment situation, surface exploration will be terminated and the robot moves the touched finger toward the edge based on the the sensed toque data as is shown in the relevant block of flowchart in Figure

4.7. In next section, the performance of algorithm is demonstrated by simulation in Matlab.

Simulation Results

In this section we present the results of algorithm with simulation of object surface in Matlab. The linear and angular velocities are shown in Figure 4.9 and 4.10. Tactile force and torque profile can be seen in Figure 4.11 and 4.12. The feeling force and misalignment angles can be found in Figure 4.13 and 4.14.

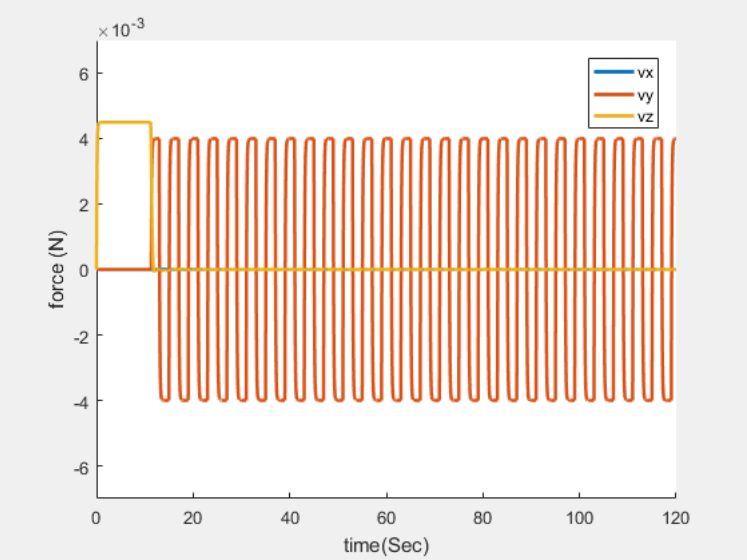

Figure 4.9: Linear velocity

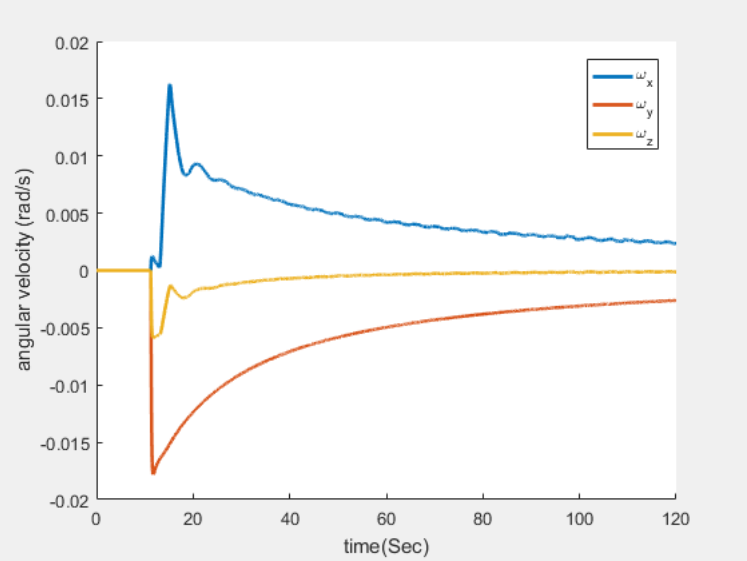

Figure 4.10: Angular velocity

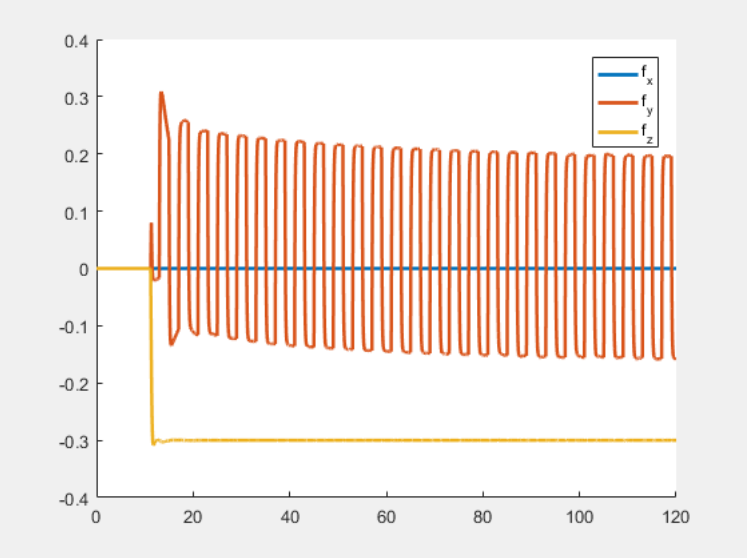

Figure 4.11: Tactile forces

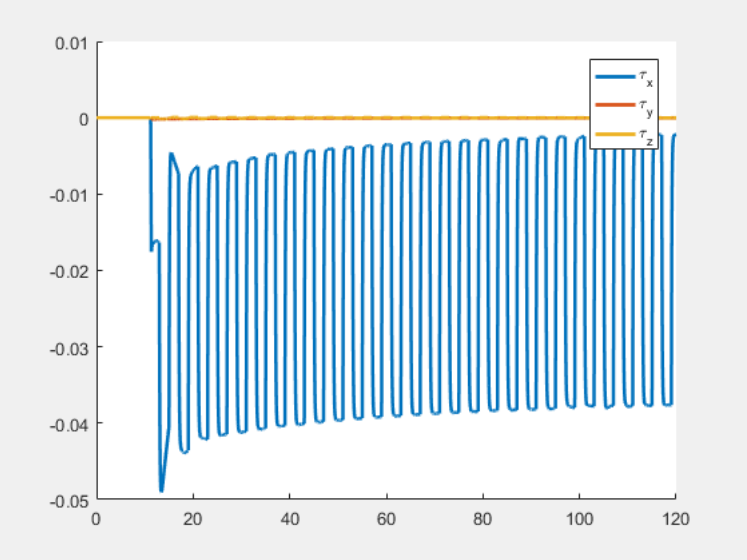

Figure 4.12: Tactile torques

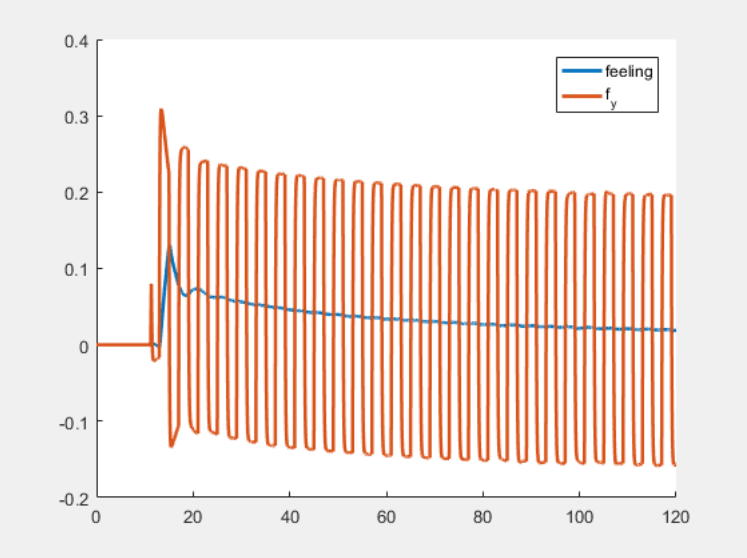

Figure 4.13: Feeling force

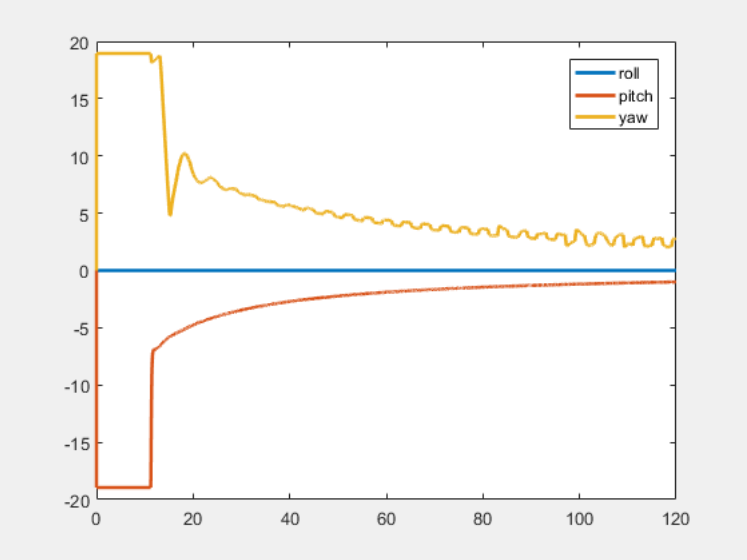

Figure 4.14: Angles

In next section, we provide the experimental results using the wrist mounted force/torque sensor to report interaction tactile data.

Experimental Results

We present the performance of algorithm to satisfy closure constraint with settings and demonstrations in this section. The robot agent for this purpose is Mico robot and we consider both rectangular and cylindrical graspable objects to validate the algorithm.

Settings

The experiments are conducted for three different initial configurations respect to the object. We present the interaction sensed force and torque and feeling force as well as robot hand linear and

angular velocities and roll, pitch and yaw angle during interaction for each demonstration as is shown in Figure 3.6.

Because the algorithm is independent of torque for alignment with object, we consider $f_{dz} = -0.3$ to achieve a touched force level. The threshold of feeling force is selected on $Th_f = 0.02N$. The parameters used in the experiment and the algorithm flowchart 4.7 are listed in Table 4.1.

Table 4.1: Parameters

Parameter	Value
v_{dz}	0.0055
f_{dz}	-0.3
Th_f	0.02
$\alpha_{\omega x}$	0.125
$\alpha_{\omega y}$	0.03
v_{dy}	0.005
β	120

Demonstrations

Rectangular object:

The candidate object for rectangular object is shown in Figure 4.15.

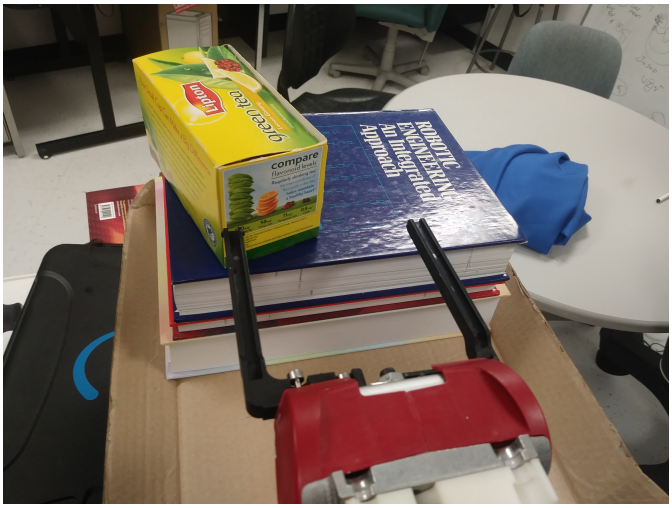

Figure 4.15: Rectangular object candidate for grasping

1. Yaw misalignment angle with surface $18.43°$ and there is no pitch misalignment angle. The video of the experiment is available at grasping rectangular 1. Tactile data during grasping the object is shown in Figures 4.16 and 4.17. Linear and angular velocities can be found in Figures 4.18 and 4.19. The time profile of pitch, roll and yaw angles is represented in Figures 4.20 as well as the feeling force in Figure 4.21.

The touched force reaches to the desired $-0.3N$ level and the same time the sensed force f_y is decreasing over time in Figure 4.16.

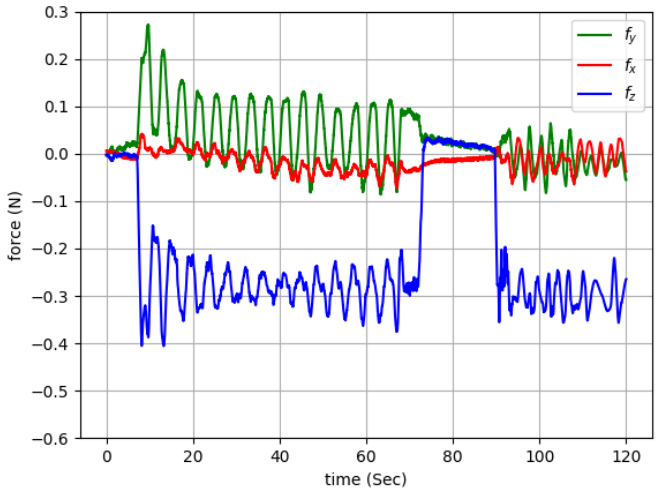

Figure 4.16: Tactile forces

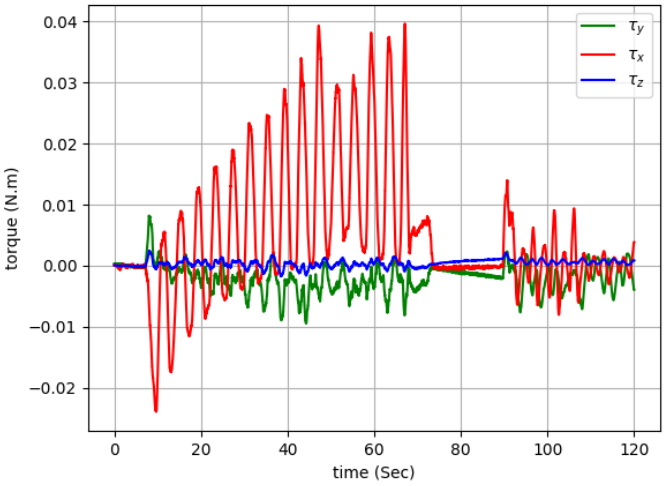

Figure 4.17: Tactile torques

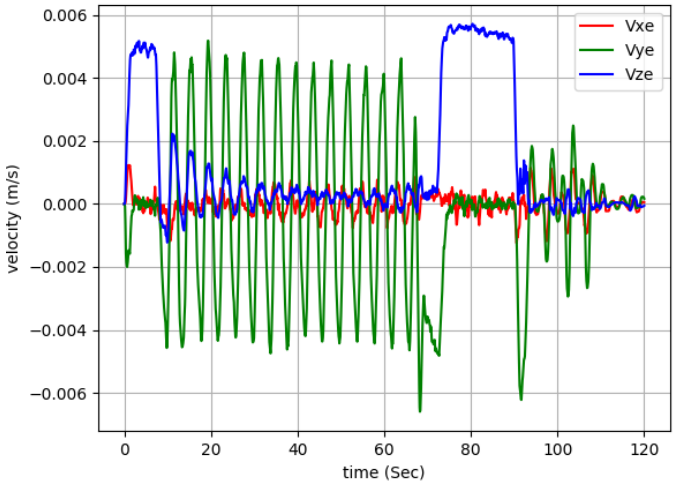

Figure 4.18: Linear velocities

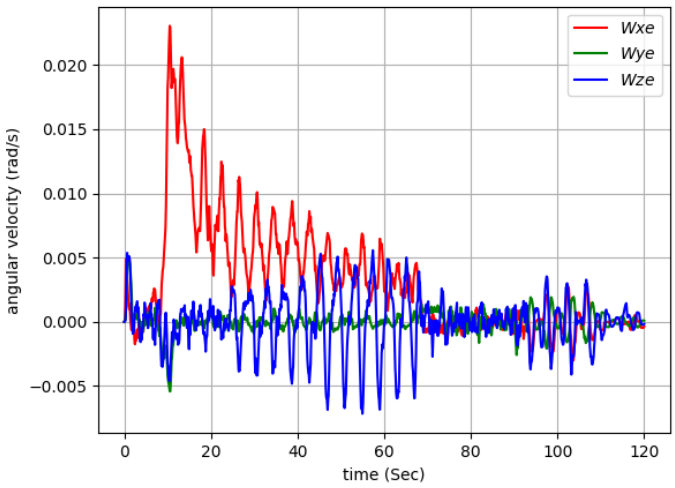

Figure 4.19: Angular velocities

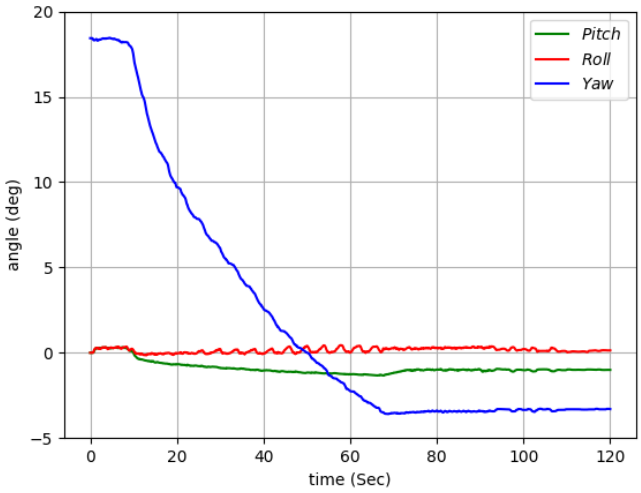

Figure 4.20: Angles

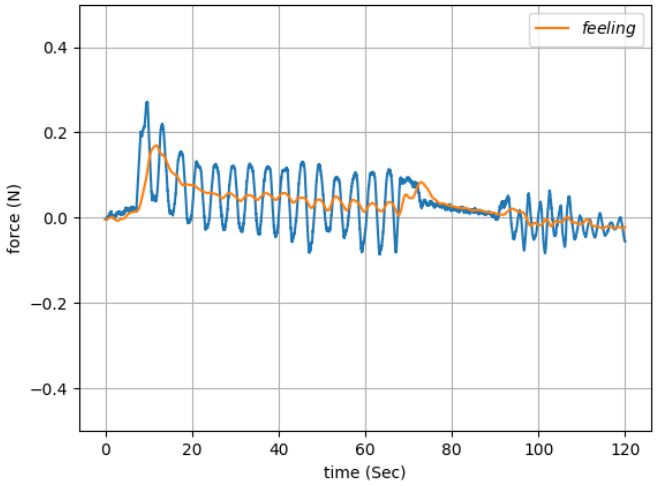

Figure 4.21: Feeling force

Considering the feeling force threshold as well as an error in ground truth, the angles converge

to the vicinity of zero in Figure 4.20. Meanwhile, the feeling force decays exponentially during perception in Figure 4.21.

2. Yaw misalignment angle with surface $-18.43°$ and pitch misalignment angle $-16.7°$. The video of the experiment is available at grasping rectangular 2. Tactile data during grasping the object is shown in Figures 4.22 and 4.23. Linear and angular velocities can be found in Figures 4.24 and 4.25. The time profile of pitch, roll and yaw angles is represented in Figures 4.26 as well as the feeling force in Figure 4.27.

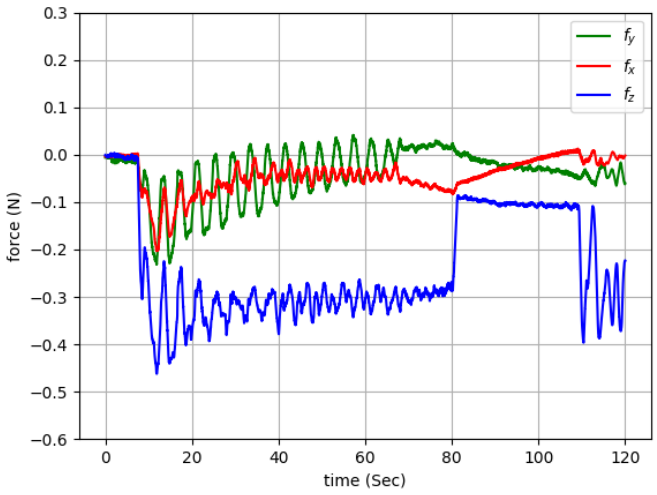

Figure 4.22: Tactile forces

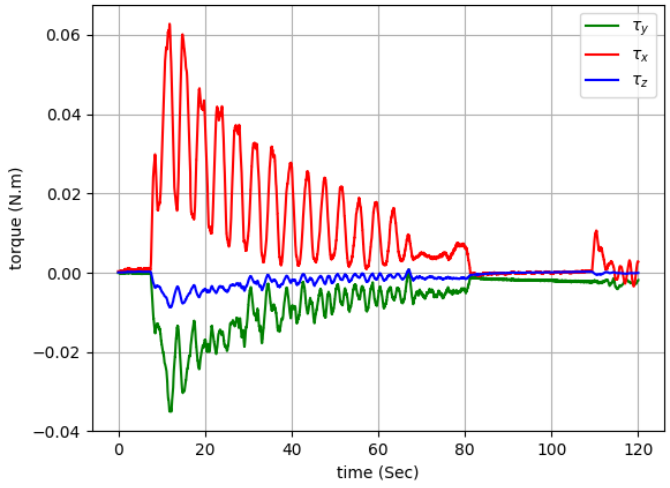

Figure 4.23: Tactile torques

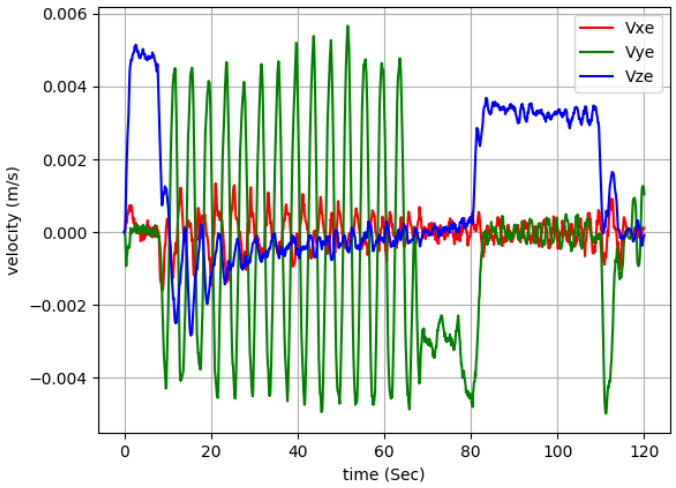

Figure 4.24: Linear velocities

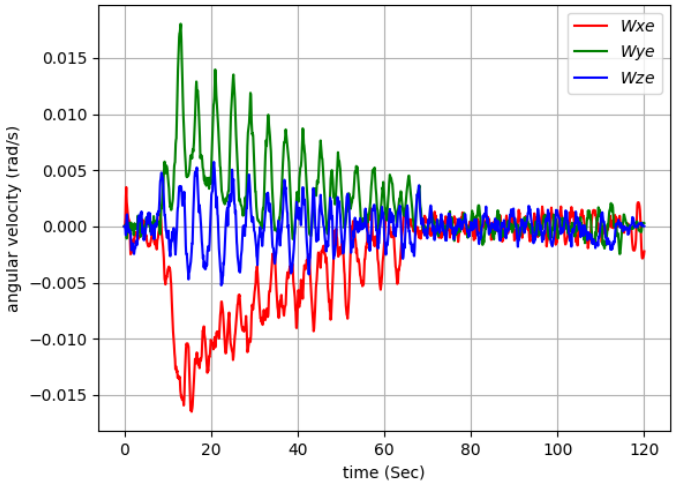

Figure 4.25: Angular velocities

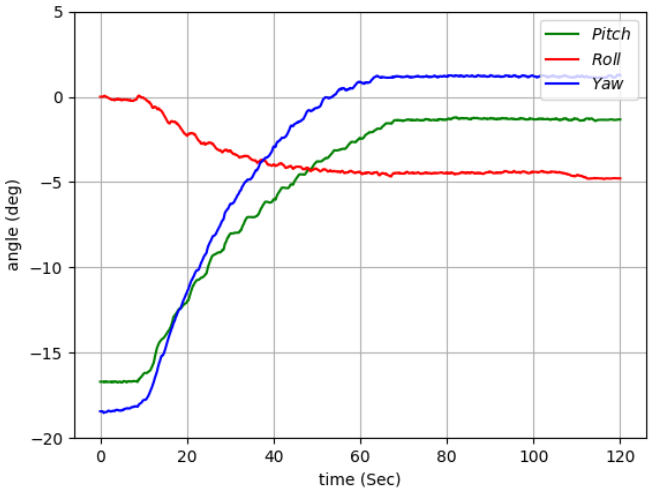

Figure 4.26: Angles

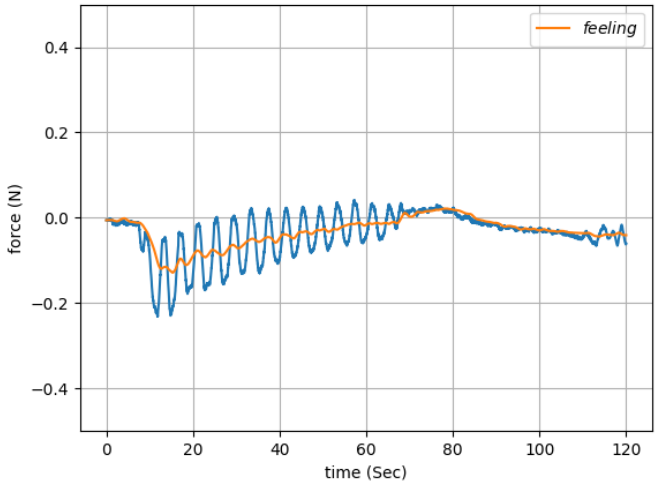

Figure 4.27: Feeling force

3. Yaw misalignment angle with surface $18.43°$ and pitch misalignment angle $-16.8°$. The video of the experiment is available at grasping rectangular 3. Tactile data during grasping the object is shown in Figures 4.28 and 4.29. Linear and angular velocities can be found in Figures 4.30 and 4.31. The time profile of pitch, roll and yaw angles is represented in Figures 4.32 as well as the feeling force in Figure 4.33.

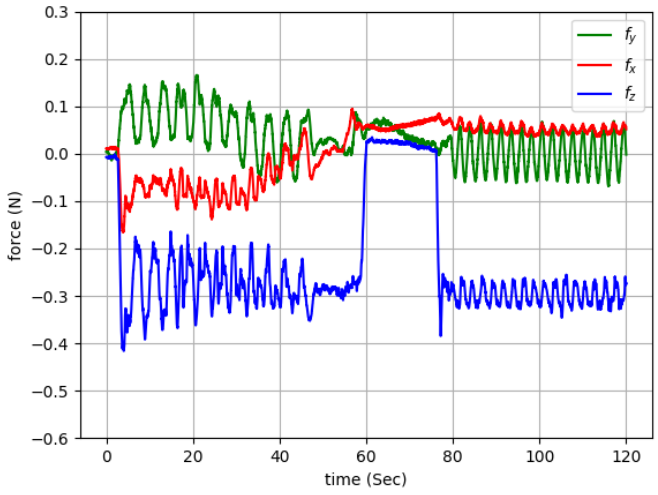

Figure 4.28: Tactile forces

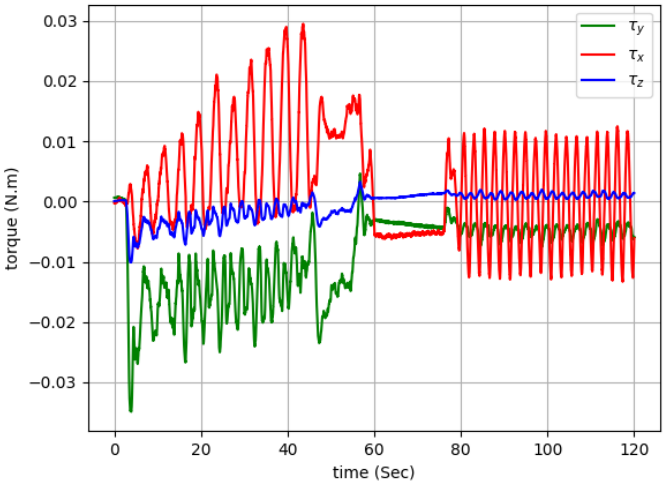

Figure 4.29: Tactile torques

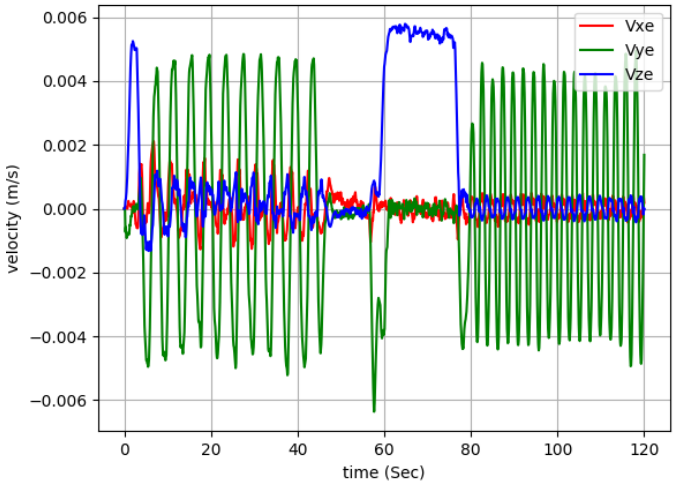

Figure 4.30: Linear velocities

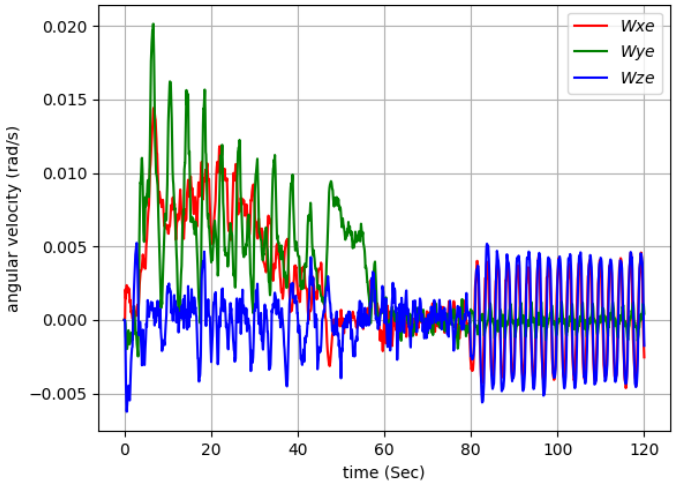

Figure 4.31: Angular velocities

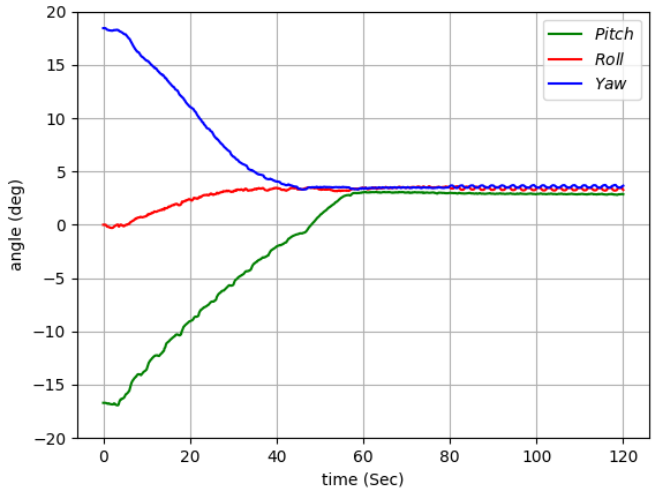

Figure 4.32: Angles

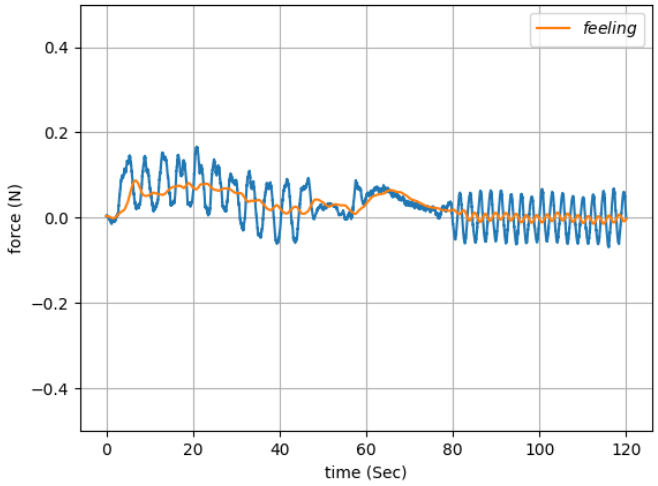

Figure 4.33: Feeling force

Cylindrical object:

The candidate object for cylindrical object is in Figure 3.14.

1. Yaw misalignment angle with surface 11.21° and there is no pitch misalignment angle. The video of the experiment is available at grasping cylindrical 1. Tactile data during grasping the object is shown in Figures 4.34 and 4.35. Linear and angular velocities can be found in Figures 4.36 and 4.37. The time profile of pitch, roll and yaw angles is represented in Figures 4.38 as well as the feeling force in Figure 4.39.

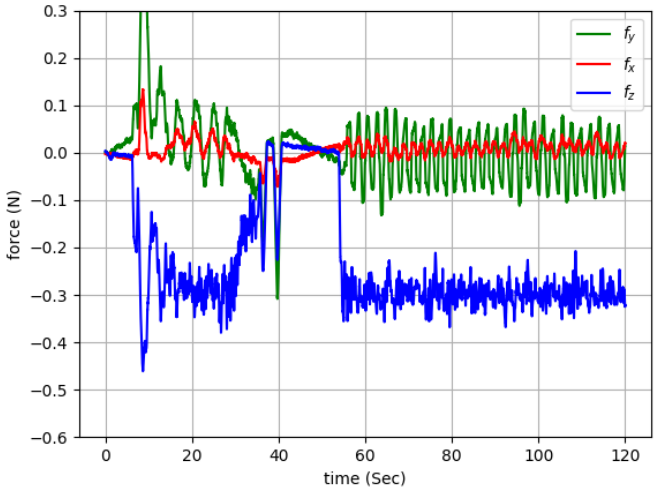

Figure 4.34: Tactile forces

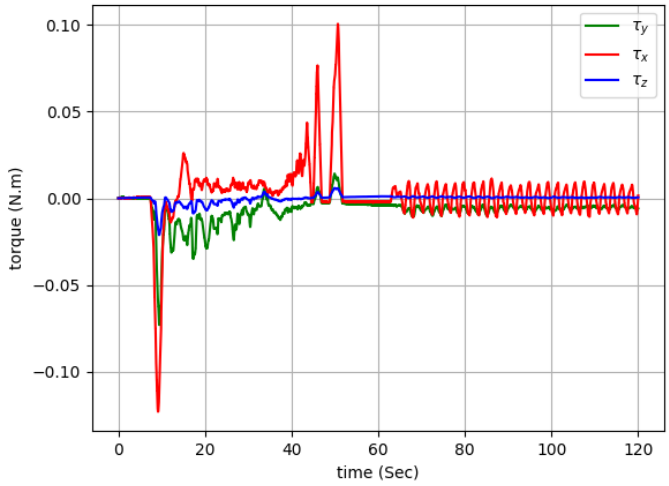

Figure 4.35: Tactile torques

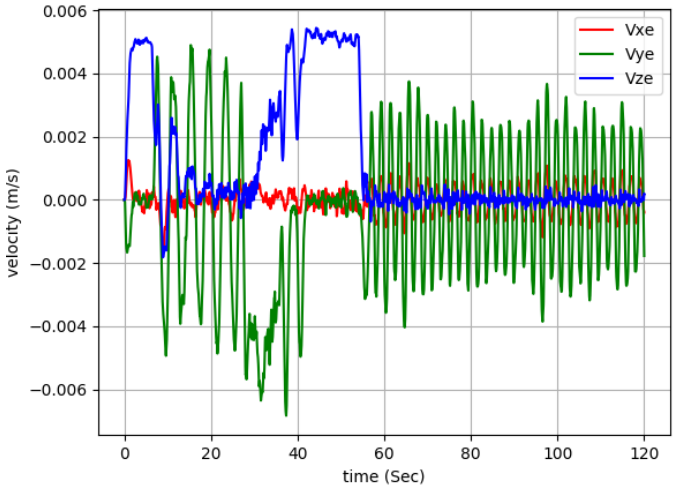

Figure 4.36: Linear velocities

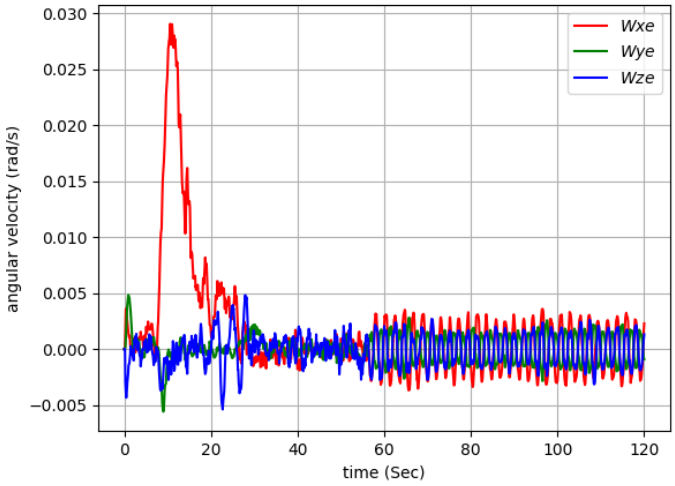

Figure 4.37: Angular velocities

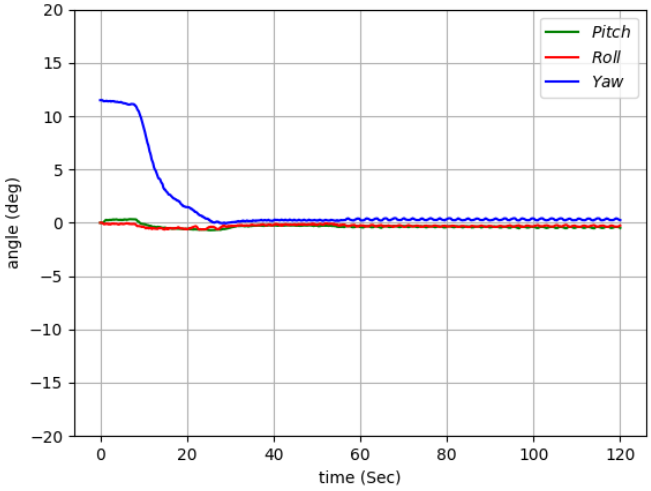

Figure 4.38: Angles

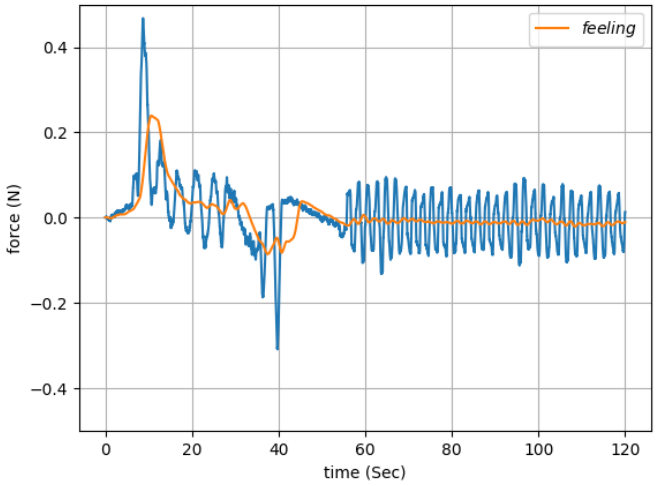

Figure 4.39: Feeling force

2. Yaw misalignment angle with surface $11.21°$ and there is pitch misalignment angle $-18.43°$. The video of the experiment is available at grasping cylindrical 2. Tactile data during grasping the object is shown in Figures 4.40 and 4.41. Linear and angular velocities can be found in Figures 4.42 and 4.43. The time profile of pitch, roll and yaw angles is represented in Figures 4.44 as well as the feeling force in Figure 4.45.

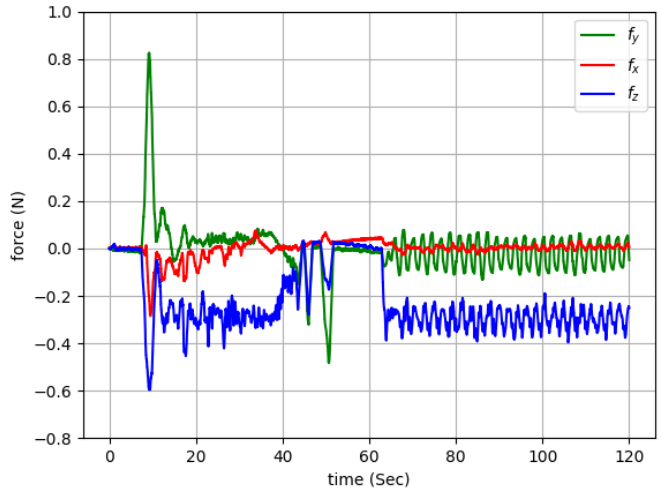

Figure 4.40: Tactile forces

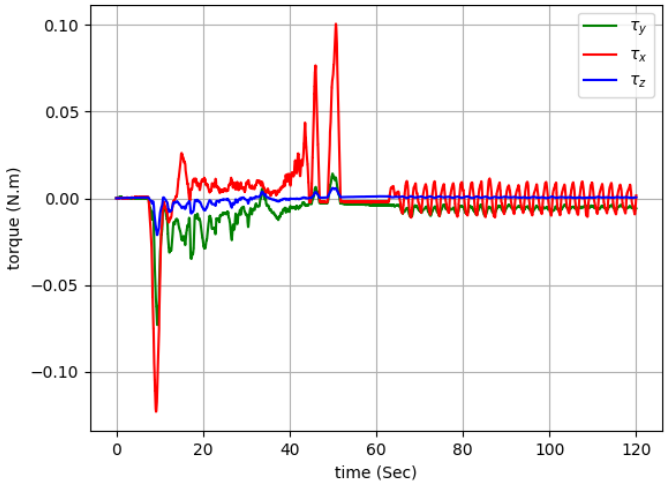

Figure 4.41: Tactile torques

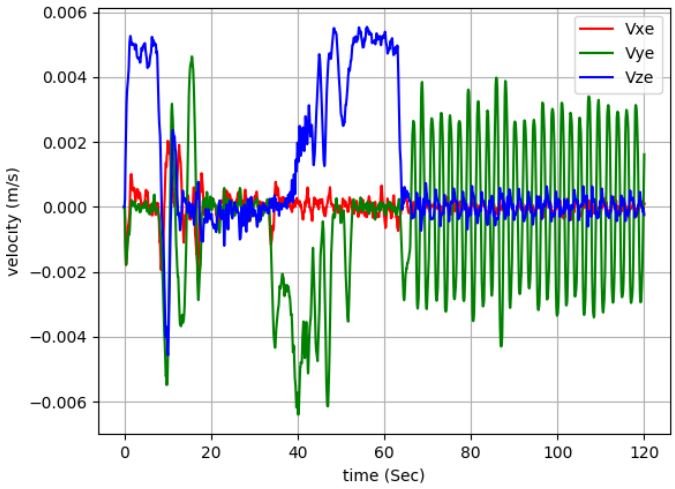

Figure 4.42: Linear velocities

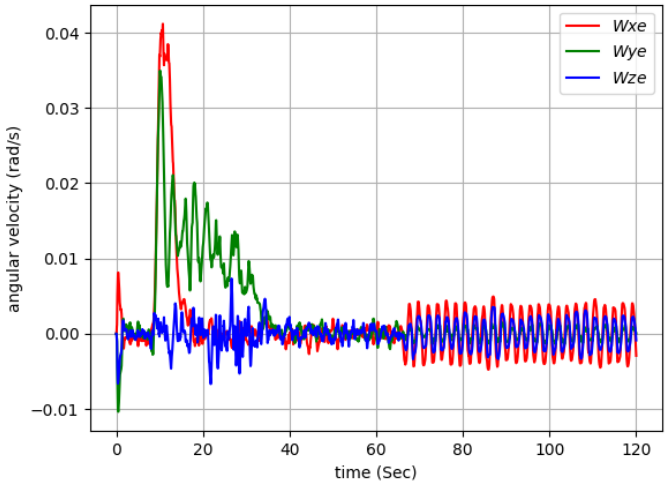

Figure 4.43: Angular velocities

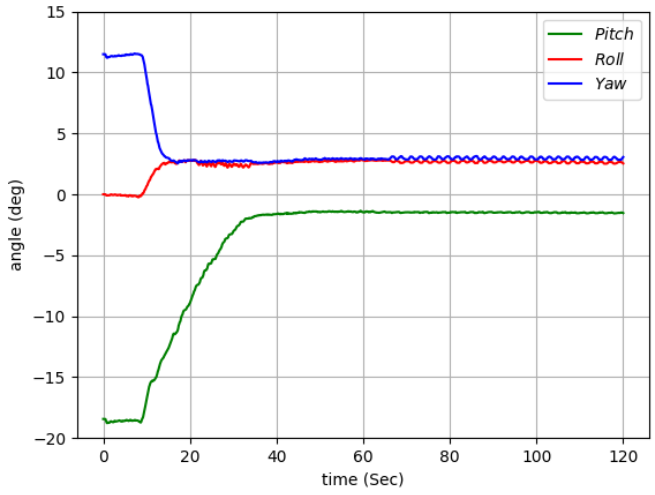

Figure 4.44: Angles

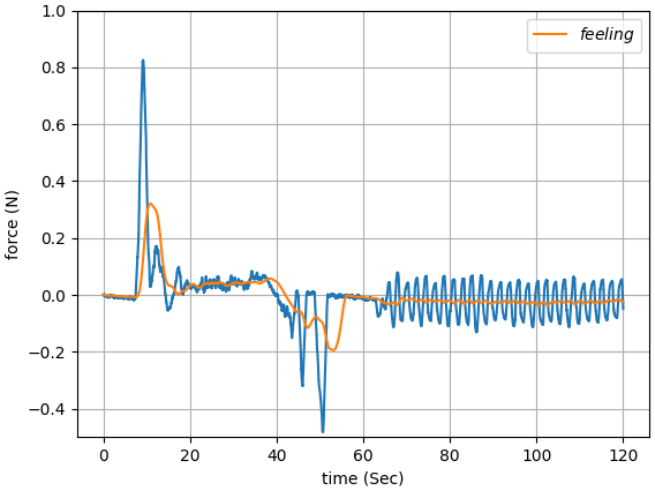

Figure 4.45: Feeling force

3. Yaw misalignment angle with surface $-7.7°$ and there is pitch misalignment angle $-14.23°$.

The video of the experiment is available at grasping cylindrical 3. Tactile data during grasping the object is shown in Figures 4.46 and 4.47.

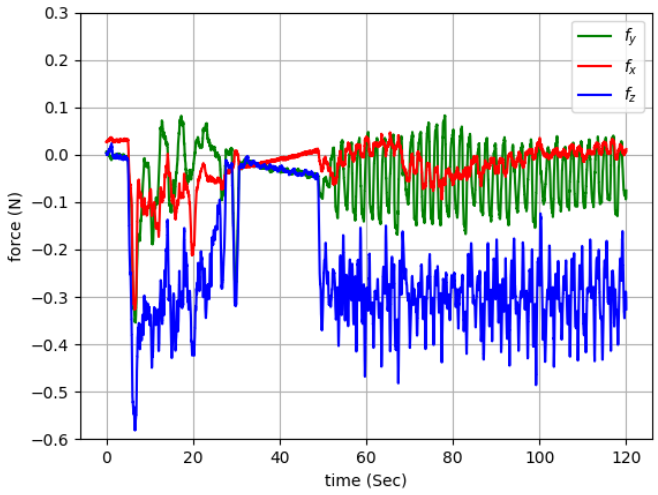

Figure 4.46: Tactile forces

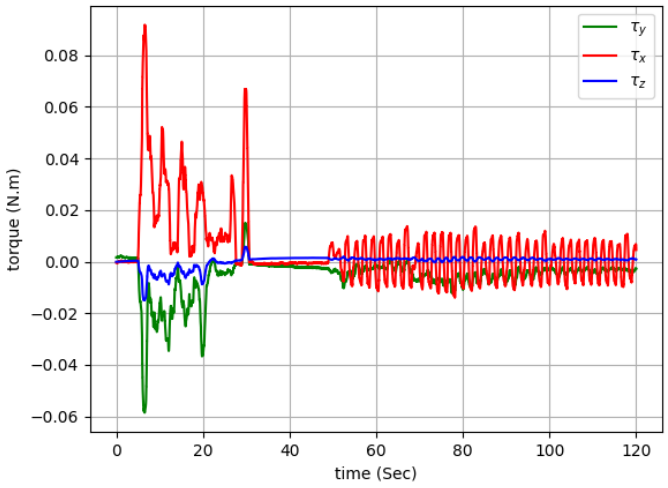

Figure 4.47: Tactile torques

Linear and angular velocities can be found in Figures 4.48 and 4.49.

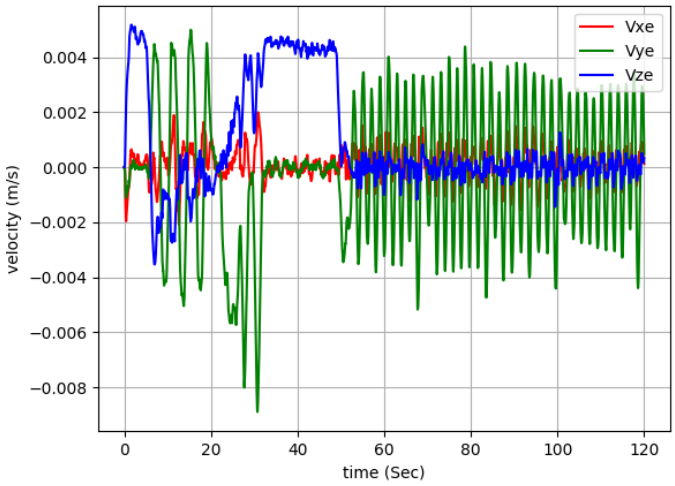

Figure 4.48: Linear velocities

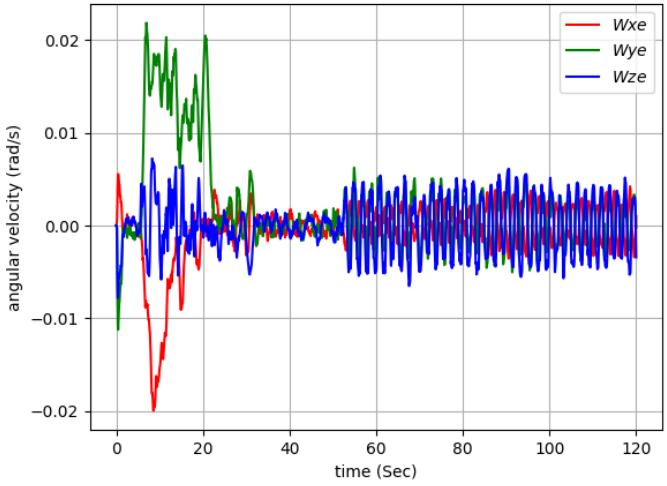

Figure 4.49: Angular velocities

The time profile of pitch, roll and yaw angles is represented in Figures 4.50 as well as the feeling force in Figure 4.51.

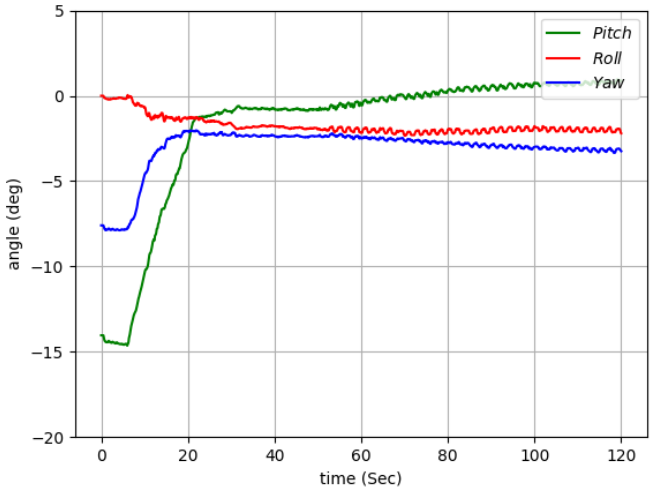

Figure 4.50: Angles

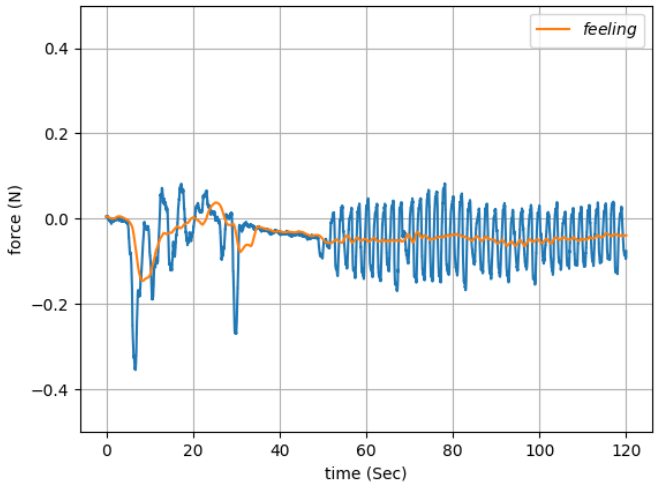

Figure 4.51: Feeling force

From the recorded data related to each experiments, the robot perceives the object after contacting with it from an initial configuration. The perception process is an interactive perception-action using real-time feeling data sensed in finger tip.

CHAPTER 5: CONCLUSION

In this dissertation, we have proposed two human-like grasping for novel objects using tactile sensing without vision. In the first algorithm, the tactile sensing data is extracted from the robot wrist sensor and calibration process is needed to rid the tactile data of the bias related to robot hand weight connected to the sensor. To compensate for the bias, we provided the mathematical equations. A grasping algorithm is suggested using this tactile data. In the second one, we consider closure constraint for object configurations that are hard to grasp for the robot by the first algorithm. A real-time interactive perception-action process affects the object feeling into grasping process similar to a human. The suggested grasping algorithm is applied to grasp a set of unknown objects and the time profile of tactile data is presented during execution of grasping.

The advantages of the proposed algorithms can be listed as follows. We showed that it is feasible to propose a more human-like grasping using tactile perception in real-time. We present the procedure to eliminate the unwanted bias data from the wrist sensor in order to successfully mine the tactile perception. We propose an approach to make sure that the robot is compliant during interaction with the object using feedback tactile data as well as minimizing the implementation of sequential logical rules. We utilized more freedom of actions reducing the dependency between them during grasping of an object for the robot. The approach perceives the object in an interactive manner in order to have a more reliable grasping. We develop the algorithm to generalize to a broader range of graspable objects. Our approach is a model free approach without prior knowledge of object model since we do not assume knowledge of object shape or impose any constraint on the shape of the object.

Our proposed algorithms have their limitations. We considered several parameters to improve the flexibility of the algorithms. Tuning these parameters can be challenging and needs time and effort

before implementing the algorithm in real scenarios. We also need to improve the algorithm in interaction with highly unstructured objects.

Interactive perception is a unique characteristics of embodied agents useful in understanding their surrounding environment. One direction for future work is to propose enhanced strategies in exploiting more freedom of robot finger actions using this ability for harder tasks. These tasks can be related to multi-finger tasks such as object recognition hold inside the fingers or changing the orientation of object for a better grasping situation before seizing the object inside the fingers. Another direction can be to perform the task according to user preferences by learning from demonstrations. Moreover, it is an interesting idea to train the robot to make decisions on how to grasp very general objects based on the provided primitive commands in this work.

LIST OF REFERENCES

[1] M. Baghbahari, N. Hajiakhoond, and A. Behal, "Real-time policy generation and its application to robot grasping," *arXiv preprint arXiv:1808.05244*, 2018.

[2] Y. Lin and Y. Sun, "Robot grasp planning based on demonstrated grasp strategies," *The International Journal of Robotics Research*, vol. 34, no. 1, pp. 26–42, 2015.

[3] A. I. Aviles, A. Marban, P. Sobrevilla, J. Fernandez, and A. Casals, "A recurrent neural network approach for 3d vision-based force estimation," in *2014 4th International Conference on Image Processing Theory, Tools and Applications (IPTA)*. IEEE, 2014, pp. 1–6.

[4] A. I. Aviles, S. Alsaleh, P. Sobrevilla, and A. Casals, "Sensorless force estimation using a neuro-vision-based approach for robotic-assisted surgery," in *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)*. IEEE, 2015, pp. 86–89.

[5] N. Haouchine, W. Kuang, S. Cotin, and M. Yip, "Vision-based force feedback estimation for robot-assisted surgery using instrument-constrained biomechanical three-dimensional maps," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2160–2165, 2018.

[6] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 421–436, 2018.

[7] H. B. Amor, O. Kroemer, U. Hillenbrand, G. Neumann, and J. Peters, "Generalization of human grasping for multi-fingered robot hands," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 2043–2050.

[8] A. Herzog, P. Pastor, M. Kalakrishnan, L. Righetti, J. Bohg, T. Asfour, and S. Schaal, "Learning of grasp selection based on shape-templates," *Autonomous Robots*, vol. 36, no. 1-2, pp. 51–65, 2014.

[9] M. Kopicki, R. Detry, M. Adjigble, R. Stolkin, A. Leonardis, and J. L. Wyatt, "One-shot learning and generation of dexterous grasps for novel objects," *The International Journal of Robotics Research*, vol. 35, no. 8, pp. 959–976, 2016.

[10] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[11] C. Goldfeder, M. Ciocarlie, J. Peretzman, H. Dang, and P. K. Allen, "Data-driven grasping with partial sensor data," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 1278–1283.

[12] M. Gualtieri, A. Ten Pas, K. Saenko, and R. Platt, "High precision grasp pose detection in dense clutter," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 598–605.

[13] E. Johns, S. Leutenegger, and A. J. Davison, "Deep learning a grasp function for grasping under gripper pose uncertainty," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4461–4468.

[14] D. Kappler, J. Bohg, and S. Schaal, "Leveraging big data for grasp planning," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 4304–4311.

[15] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 1316–1322.

[16] N. Hudson, T. Howard, J. Ma, A. Jain, M. Bajracharya, S. Myint, C. Kuo, L. Matthies, P. Backes, P. Hebert *et al.*, "End-to-end dexterous manipulation with deliberate interactive estimation," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 2371–2378.

[17] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 3406–3413.

[18] J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, and K. Goldberg, "Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1957–1964.

[19] J. Oberlin and S. Tellex, "Autonomously acquiring instance-based object models from experience," in *Robotics Research*. Springer, 2018, pp. 73–90.

[20] B. Kehoe, A. Matsukawa, S. Candido, J. Kuffner, and K. Goldberg, "Cloud-based robot grasping with the google object recognition engine," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 4263–4270.

[21] B. Kehoe, S. Patil, P. Abbeel, and K. Goldberg, "A survey of research on cloud robotics and automation," *IEEE Transactions on automation science and engineering*, vol. 12, no. 2, pp. 398–409, 2015.

[22] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesis—a survey," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289–309, 2013.

[23] L. Jamone, M. Brandao, L. Natale, K. Hashimoto, G. Sandini, and A. Takanishi, "Autonomous online generation of a motor representation of the workspace for intelligent whole-body reaching," *Robotics and Autonomous Systems*, vol. 62, no. 4, pp. 556–567, 2014.

[24] L. Jamone, L. Natale, F. Nori, G. Metta, and G. Sandini, "Autonomous online learning of reaching behavior in a humanoid robot," *International Journal of Humanoid Robotics*, vol. 9, no. 03, p. 1250017, 2012.

[25] D. Kragic, H. I. Christensen *et al.*, "Survey on visual servoing for manipulation," *Computational Vision and Active Perception Laboratory, Fiskartorpsv*, vol. 15, p. 2002, 2002.

[26] P. Hebert, N. Hudson, J. Ma, T. Howard, T. Fuchs, M. Bajracharya, and J. Burdick, "Combined shape, appearance and silhouette for simultaneous manipulator and object tracking," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 2405–2412.

[27] K. Mohta, V. Kumar, and K. Daniilidis, "Vision based control of a quadrotor for perching on planes and lines," in *International Conference on Robotics and Automation (ICRA)*, 2014.

[28] K. Hosoda and M. Asada, "Versatile visual servoing without knowledge of true jacobian," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94)*, vol. 1. IEEE, 1994, pp. 186–193.

[29] S. Koo and S. Behnke, "Focused online visual-motor coordination for a dual-arm robot manipulator," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1579–1586.

[30] F. Widmaier, D. Kappler, S. Schaal, and J. Bohg, "Robot arm pose estimation by pixel-wise regression of joint angles," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 616–623.

[31] S. Chitta, J. Sturm, M. Piccoli, and W. Burgard, "Tactile sensing for mobile manipulation," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 558–568, 2011.

[32] A. G. Eguíluz, I. Rano, S. A. Coleman, and T. M. McGinnity, "Reliable object handover through tactile force sensing and effort control in the shadow robot hand," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 372–377.

[33] A. Cirillo, P. Cirillo, G. De Maria, C. Natale, and S. Pirozzi, "Control of linear and rotational slippage based on six-axis force/tactile sensor," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 1587–1594.

[34] D. Guo, F. Sun, H. Liu, T. Kong, B. Fang, and N. Xi, "A hybrid deep architecture for robotic grasp detection," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 1609–1614.

[35] Y. Gao, L. A. Hendricks, K. J. Kuchenbecker, and T. Darrell, "Deep learning for tactile understanding from visual and haptic data," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 536–543.

[36] H. van Hoof, N. Chen, M. Karl, P. van der Smagt, and J. Peters, "Stable reinforcement learning with autoencoders for tactile and visual data," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 3928–3934.

[37] C. Finn, X. Y. Tan, Y. Duan, T. Darrell, S. Levine, and P. Abbeel, "Deep spatial autoencoders for visuomotor learning," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 512–519.

[38] Y. Chebotar, O. Kroemer, and J. Peters, "Learning robot tactile sensing for object manipulation," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 3368–3375.

[39] Y. Chebotar, K. Hausman, Z. Su, G. S. Sukhatme, and S. Schaal, "Self-supervised regrasping using spatio-temporal tactile features and reinforcement learning," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 1960–1966.

[40] O. Kroemer, C. H. Lampert, and J. Peters, "Learning dynamic tactile sensing with robust -based training," *IEEE transactions on robotics*, vol. 27, no. 3, pp. 545–557, 2011.

[41] A. Schneider, J. Sturm, C. Stachniss, M. Reisert, H. Burkhardt, and W. Burgard, "Object identification with tactile sensors using bag-of-features." in *IROS*, vol. 9, 2009, pp. 243–248.

[42] A. Petrovskaya, O. Khatib, S. Thrun, and A. Y. Ng, "Touch based perception for object manipulation."

[43] C. Corcoran and R. Platt, "A measurement model for tracking hand-object state during dexterous manipulation," in *2010 IEEE International Conference on Robotics and Automation*. IEEE, 2010, pp. 4302–4308.

[44] R. Li, R. Platt, W. Yuan, A. ten Pas, N. Roscup, M. A. Srinivasan, and E. Adelson, "Localization and manipulation of small parts using gelsight tactile sensing," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 3988–3993.

[45] Y. Bekiroglu, J. Laaksonen, J. A. Jørgensen, V. Kyrki, and D. Kragic, "Assessing grasp stability based on learning and haptic data," *IEEE Transactions on Robotics*, vol. 27, no. 3, p. 616, 2011.

[46] H. Dang and P. K. Allen, "Stable grasping under pose uncertainty using tactile feedback," *Autonomous Robots*, vol. 36, no. 4, pp. 309–330, 2014.

[47] D. A. Nowak, S. Glasauer, and J. Hermsdörfer, "Force control in object manipulation—a model for the study of sensorimotor control strategies," *Neuroscience & Biobehavioral Reviews*, vol. 37, no. 8, pp. 1578–1586, 2013.

[48] J. Bohg, K. Hausman, B. Sankaran, O. Brock, D. Kragic, S. Schaal, and G. S. Sukhatme, "Interactive perception: Leveraging action in perception and perception in action," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1273–1291, 2017.

[49] J. Schill, J. Laaksonen, M. Przybylski, V. Kyrki, T. Asfour, and R. Dillmann, "Learning continuous grasp stability for a humanoid robot hand based on tactile sensing," in *2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*. IEEE, 2012, pp. 1901–1906.

[50] M. Li, Y. Bekiroglu, D. Kragic, and A. Billard, "Learning of grasp adaptation through experience and tactile sensing," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 3339–3346.

[51] A. Bicchi, M. Bergamasco, P. Dario, and A. Fiorillo, "Integrated tactile sensing for gripper fingers," in *Int. Conf. on Robot Vision and Sensory Control*, 1988.

[52] J. M. Romano, K. Hsiao, G. Niemeyer, S. Chitta, and K. J. Kuchenbecker, "Human-inspired robotic grasp control with tactile sensing," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1067–1079, 2011.

[53] F. Veiga, H. Van Hoof, J. Peters, and T. Hermans, "Stabilizing novel objects by learning to predict tactile slip," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 5065–5072.

[54] P. K. A. A. T. Miller and P. Y. O. B. S. Leibowitz, "Integration of vision, force and tactile sensing for grasping," *Int. J. Intell. Mach*, vol. 4, pp. 129–149, 1999.

[55] Y. Bekiroglu, "Learning to assess grasp stability from vision, touch and proprioception," Ph.D. dissertation, KTH Royal Institute of Technology, 2012.

[56] C. Jara, J. Pomares, F. Candelas, and F. Torres, "Control framework for dexterous manipulation using dynamic visual servoing and tactile sensors' feedback," *Sensors*, vol. 14, no. 1, pp. 1787–1804, 2014.

[57] Y. Bekiroglu, A. Damianou, R. Detry, J. A. Stork, D. Kragic, and C. H. Ek, "Probabilistic consolidation of grasp experience," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 193–200.

[58] D. Guo, F. Sun, B. Fang, C. Yang, and N. Xi, "Robotic grasping using visual and tactile sensing," *Information Sciences*, vol. 417, pp. 274–286, 2017.

[59] E. Hyttinen, D. Kragic, and R. Detry, "Estimating tactile data for adaptive grasping of novel objects," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*. IEEE, 2017, pp. 643–648.

[60] R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, and S. Levine, "More than a feeling: Learning to grasp and regrasp using vision and touch," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3300–3307, 2018.

[61] F. R. Hogan, M. Bauza, O. Canal, E. Donlon, and A. Rodriguez, "Tactile regrasp: Grasp adjustments via simulated tactile transformations," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 2963–2970.

[62] A. Murali, Y. Li, D. Gandhi, and A. Gupta, "Learning to grasp without seeing," *arXiv preprint arXiv:1805.04201*, 2018.

[63] P. Sikka, H. Zhang, and S. Sutphen, "Tactile servo: Control of touch-driven robot motion," in *Experimental Robotics III*. Springer, 1994, pp. 219–233.

[64] Q. Li, C. Schürmann, R. Haschke, and H. Ritter, "A control framework for tactile servoing," 2013.

[65] S. Tian, F. Ebert, D. Jayaraman, M. Mudigonda, C. Finn, R. Calandra, and S. Levine, "Manipulation by feel: Touch-based control with deep predictive models," *arXiv preprint arXiv:1903.04128*, 2019.

[66] A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 1. IEEE, 2000, pp. 348–353.

[67] C. Borst, M. Fischer, and G. Hirzinger, "Grasp planning: How to choose a suitable task wrench space," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, vol. 1. IEEE, 2004, pp. 319–325.

[68] D. Berenson, R. Diankov, K. Nishiwaki, S. Kagami, and J. Kuffner, "Grasp planning in complex scenes," in *2007 7th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2007, pp. 42–48.

[69] M. T. Ciocarlie and P. K. Allen, "Hand posture subspaces for dexterous robotic grasping," *The International Journal of Robotics Research*, vol. 28, no. 7, pp. 851–867, 2009.

[70] A. Sahbani, S. El-Khoury, and P. Bidaud, "An overview of 3d object grasp synthesis algorithms," *Robotics and Autonomous Systems*, vol. 60, no. 3, pp. 326–336, 2012.

[71] A. ten Pas and R. Platt, "Using geometry to detect grasp poses in 3d point clouds," in *Robotics Research*. Springer, 2018, pp. 307–324.

[72] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," *arXiv preprint arXiv:1703.09312*, 2017.

[73] E. Luberto, Y. Wu, G. Santaera, M. Gabiccini, and A. Bicchi, "Enhancing adaptive grasping through a simple sensor-based reflex mechanism," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1664–1671, 2017.

[74] H. Yousef, M. Boukallel, and K. Althoefer, "Tactile sensing for dexterous in-hand manipulation in robotics—a review," *Sensors and Actuators A: physical*, vol. 167, no. 2, pp. 171–187, 2011.

[75] M. Stachowsky, T. Hummel, M. Moussa, and H. A. Abdullah, "A slip detection and correction strategy for precision robot grasping," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 5, pp. 2214–2226, 2016.

[76] A. Ajoudani, E. Hocaoglu, A. Altobelli, M. Rossi, E. Battaglia, N. Tsagarakis, and A. Bicchi, "Reflex control of the pisa/iit softhand during object slippage," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1972–1979.

[77] S. Dong, W. Yuan, and E. H. Adelson, "Improved gelsight tactile sensor for measuring geometry and slip," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 137–144.

[78] G. De Maria, C. Natale, and S. Pirozzi, "Force/tactile sensor for robotic applications," *Sensors and Actuators A: Physical*, vol. 175, pp. 60–72, 2012.

[79] M. C. Koval, M. R. Dogar, N. S. Pollard, and S. S. Srinivasa, "Pose estimation for contact manipulation with manifold particle filters," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 4541–4548.

[80] G. Izatt, G. Mirano, E. Adelson, and R. Tedrake, "Tracking objects with point clouds from vision and touch," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 4000–4007.

[81] P. Pastor, M. Kalakrishnan, L. Righetti, and S. Schaal, "Towards associative skill memories," in *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*. IEEE, 2012, pp. 309–315.

[82] S. M. Khansari-Zadeh and O. Khatib, "Learning potential functions from human demonstrations with encapsulated dynamic and compliant behaviors," *Autonomous Robots*, vol. 41, no. 1, pp. 45–69, 2017.

[83] M. Khansari, E. Klingbeil, and O. Khatib, "Adaptive human-inspired compliant contact primitives to perform surface–surface contact under uncertainty," *The International Journal of Robotics Research*, vol. 35, no. 13, pp. 1651–1675, 2016.

[84] M. Baghbahari and A. Behal, "Automatic grasping using tactile sensing and deep calibration," in *Proceedings of the Future Technologies Conference*. Springer, 2019, pp. 175–192.

[85] A. M. Okamura, N. Smaby, and M. R. Cutkosky, "An overview of dexterous manipulation," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 1. IEEE, 2000, pp. 255–262.